Bit

Copia riservata agli abbonati

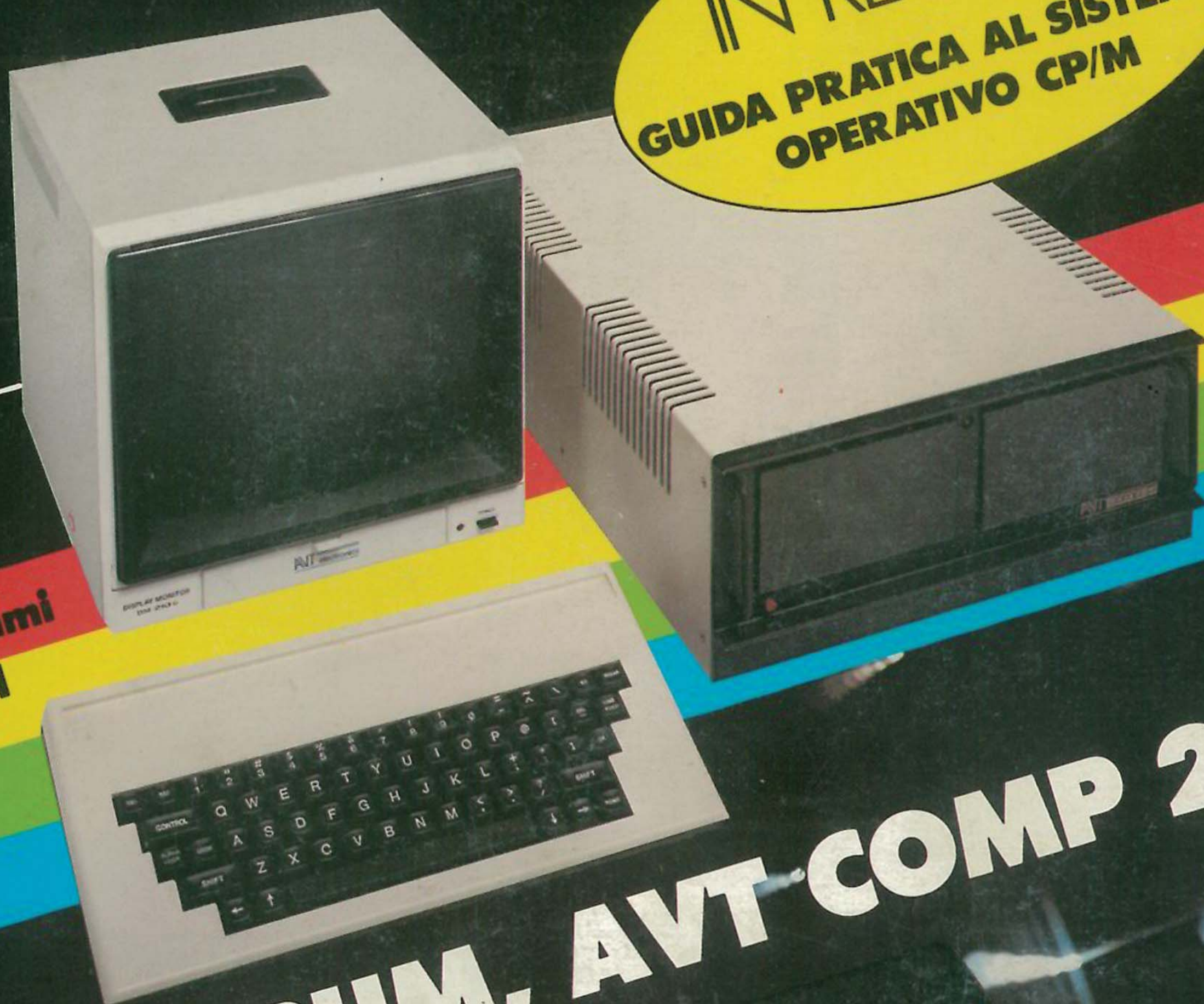
UNA PUBBLICAZIONE
DEL GRUPPO EDITORIALE JACKSON

ANNO 6 N. 38

LA PRIMA
E PIU' DIFFUSA RIVISTA
DI PERSONAL COMPUTER

L.4.000

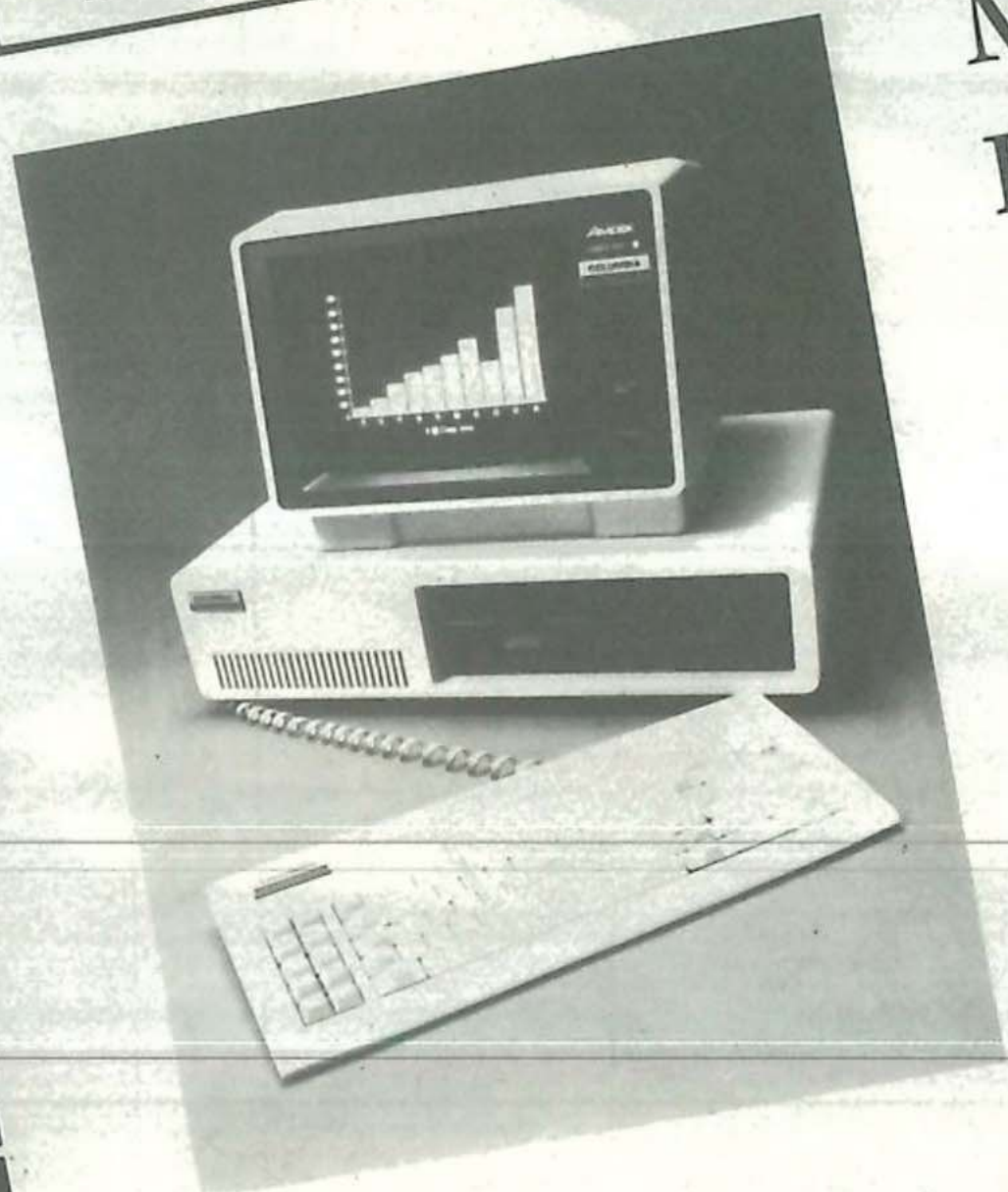
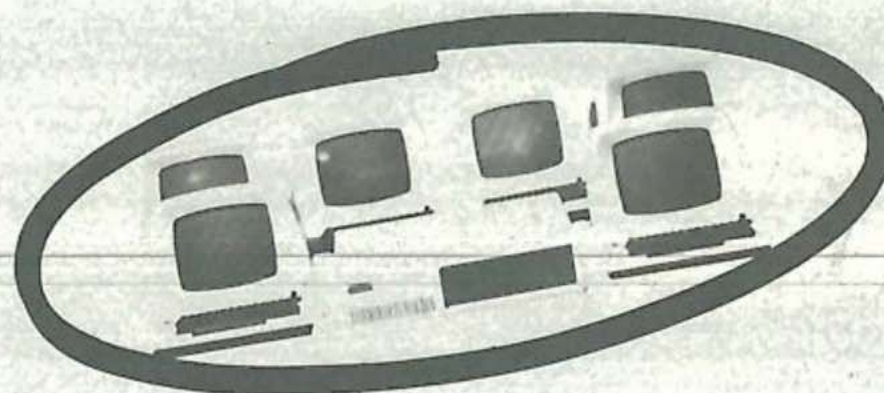
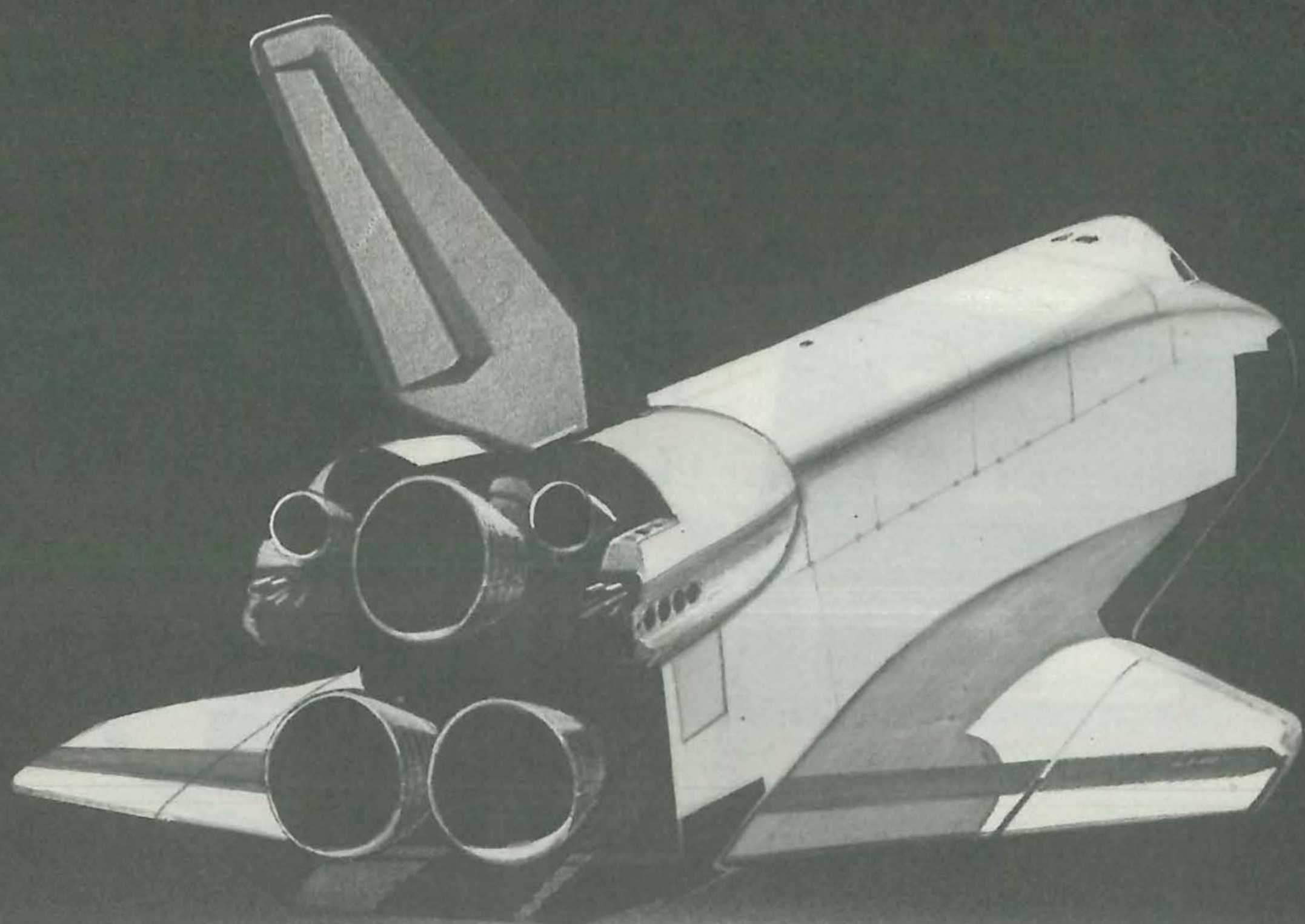
IN REGALO
GUIDA PRATICA AL SISTEMA
OPERATIVO CP/M

In questo numero
SPECIALE
RISERVATO
PERSONAL
64 pagine di programmi
per il tuo Personal
Computer

BITEST
ZX SPECTRUM, AVT COMP 2

L'AIM
PROGRAMMA
LE EPROM

AVVENTURA
NEL CASTELLO

DATASTAR

ELEDRA
PERSONAL COMPUTER
NEWS
FEBBRAIO 1983
11
PUBBLICAZIONE GRATUITA
GRUPPO ELEDRA
COLUMBIA
DATA PRODUCTS, INC.
MPC 1600
MULTI-
PERSONAL
COMPUTER
SISTEMA a 16-BIT con CPU 8088
COMPATIBILE MS-DOS, CP/M-86 e 80 e UNIX
MULTI-UTENTE COMPATIBILE MP/M-86
IBM-COMPATIBILE 3270/BISYNC/SNA/SDLC
ELEDRA 3S S.p.A. - Viale Elvezia, 18 - 20154 Milano
PUTER
GIUGNO 1982
1
Personal
IN VENDITA PRESSO I RIVENDITORI
AUTORIZZATI PERSONAL COMPUTER
ELEDRA 3S
RICHIESTA DI ABBONAMENTO GRATUITO
Spedire il coupon in busta chiusa a:
ELEDRA 3S S.p.A , - Viale Elvezia, 18 - 20154 Milano
BIT 11
☐ Desidero ricevere regolarmente Eledra Personal Computer News
☐ Ricevo già EPCN. ☐ Desidero avere informazioni su MPC COLUMBIA
☐ Indicatemi il vostro rivenditore più vicino.
Cognome e nome
Tit. Attività
Ditta
Indirizzo
CAP Città Tel. /

Arrivano i nostri.

Parliamo dei computers, o, se preferisci, di un avvenimento importante per l'efficienza della tua azienda.
Oggi infatti diventa facile applicare i computers piú affidabili, garantiti dal nome delle marche che contano di piú al mondo: con queste, la tua azienda farà un volo che la porterà avanti di anni e anni.
Vieni a Computer City: una vasta rete di centri specializzati nei personal computers, i piú adeguati alle tue necessità, dove la vendita viaggia con l'assistenza di un personale esperto e qualificato, di cui ti puoi fidare.
Tutti pionieri a Computer City.

computer city
Parla la tua lingua.

Realität

Anno 6 - n° 38 - Aprile 1983

INDICE INSERZIONISTI

In copertina:
AVT Comp 2
Sinclair Spectrum
(fotografie: Photo-In)

Sommario

DIRETTORE RESPONSABILE
Giampietro Zanga
DIRETTORE TECNICO
Gianni Giaccaglini
CAPO REDATTORE
Pietro Dell'Orco
REDAZIONE
Lorenzo Barrile
REDAZIONE USA
Sergio Mello-Grand
Gabriella Martino
GRAFICA, IMPAGINAZIONE E DISEGNI
Gianfranco De Rienzo
FOTO
Photo-In
DIFFUSIONE E ABBONAMENTI
Ombretta Giannetto,
Adela Bel Lozano,
Luigi De Cao, Marco Benedetti,
Claudio Bautti

GRUPPO EDITORIALE JACKSON Srl

DIREZIONE, REDAZIONE, AMMINISTRAZIONE
Via Rosellini, 12 - 20124 Milano
Telefoni 68.03.68 - 68.00.54
Telex 333436 GEJIT I

SEDE LEGALE
Via Vincenzo Monti, 15 - 20123 Milano

REDAZIONE USA
GEJ Publishing Group Inc.
1143 Quince Avenue
94087 Sunnyvale, CA
Tel. (408) 7730103

DIREZIONE EDITORIALE
Giampietro Zanga e Paolo Reina

COORDINAMENTO EDITORIALE
Daniele Comboni

Autorizzazione del Tribunale di Milano n. 445 del 16/12/1978

PUBBLICITA': Concessionario per l'Italia e l'Estero
Reina s.r.l.
Via Washington, 50 - 20146 Milano
Tel. (02) 4988066/7/8/9/060 (5 linee r.a.)
Telex 316213 REINA I

Concessionario pubblicità per USA e Canada:
International Media Marketing
16704 Marquardt Avenue
P.O Box 1217
Cerritos CA 90701
(213) 926-9552

STAMPA:
Reweba (Brescia)

Concessionario esclusivo per la diffusione in Italia e all'Estero:
SODIP - Via Zuretti, 25 - 20125 Milano

Spedizione in abbonamento Postale Gruppo III/70

Prezzo della rivista L. 4.000
Numero arretrato L. 6.000
Abbonamento annuo L. 35.000
per l'Estero L. 52.000
I versamenti vanno indirizzati a:
Gruppo Editoriale Jackson
Via Rosellini, 12 - 20124 Milano
mediante emissione di assegno bancario, cartolina vaglia o utilizzando il c/c Postale numero 11666203
Per i cambi di indirizzo, indicare, oltre naturalmente al nuovo, anche l'indirizzo precedente, ed allegare alla comunicazione l'importo di L. 500, anche in francobolli.

Mensile associato all'USPI
Unione Stampa Periodica Italiana

abc-26D
VDU
ai-M16
DIECINQUE, proseguendo nel proporre il meglio del mercato dell'informatica, accanto alla affermata serie dei sistemi ABC, simbolo della più alta tecnologia ed efficienza nella gestione aziendale, implementa ed espande la gamma dei suoi prodotti introducendo la nuova linea di super microcomputer a 16 Bit.
Questa nuova famiglia si basa su una struttura con più microprocessori a 16 Bit della famiglia INTEL.
La memoria principale parte da 512KB per espandersi fino ad 1MB mentre le memorie di massa collegabili sfruttano le più moderne tecnologie a Winchester per unità multiple.
Sono inoltre disponibili 16 linee per collegare più unità video-terminali, monocromatici o a colori con possibilità grafiche di grande rilievo, unità stampanti e periferiche specializzate.
Tutto ciò è supportato da vari Sistemi Operativi quali il GENIX (versione AI del noto Sistema Operativo UNIX™), nonché il CP/M 86, MP/M86, MS DOS.
DIECINQUE integra i componenti, Hardware, Software di Base e Software Applicativo, con un servizio di formazione professionale e assistenza tecnica altamente qualificato.
Diecinque
L'informatica presente e futura.
DIECINQUE S.p.A. 40069 Zola Predosa - Bologna
Via Nannetti, 1 - Tel. (051) 752264 - Telex 214848 OLCIN I
Ai Electronics Corp.
Tokyo-Japan.

Riservato Personal: e Bit si arricchisce

L'idea è stata del direttore responsabile di questa come di quasi tutte le riviste Jackson: Giampietro Zanga.
È dotato d'uno spirito e d'una souplesse degna d'un gatto. Sornione come tutti i felini, mentre in redazione ferveva inter nos il bizantineggiare attorno ad una rinnovata struttura del nostro personal magazine all'italiana, se n'è venuto fuori un bel mattino (i gatti, si sa, le pensano nel buio delle notti dei vicoli-miracoli...) con la sua proposta: 1) si fa l'inserto; *2)* lo facciamo di 64 pagine extra; *3)* ci mettiamo applicazioni.
D'ogni tipo e, meglio ancora, per uno spettro di microsistemi più ampio possibile, però, preferibilmente, di tipo professionale e d'una certa utilità. I presenti hanno subito emesso un sospiro di sollievo: si temeva il varo di una terza pubblicazione sui personal e invece ne risulta un arricchimento di **Bit** *che amplia notevolmente il numero delle sue pagine, con un logico incremento del prezzo - ahimè la carta costa! - che comunque oltre a premiare gli abbonati che non ne ricevono alcun aggravio, si mantiene entro limiti ragionevoli.*
Ho presto pensato di battezzare **Riservato Personal** *questa sorta di rivista nella rivista, a motivo del suo carattere assai privato (nello spirito di ciò che i francesi chiamano ordinateur individuel) pur nella pubblicità erga omnes della cornice. La si potrà staccare e conservare per l'uso. Ma questo non vuol certo dire che il "vecchio"* **Bit** *che racchiude e contorna l'inserto venga declassato alla categoria del radi-e-getta. All'effimero indubbio delle novità giornalistiche esso unirà sempre più - approfittando del maggior spazio lasciato libero dai defunti-traslocati Apple Club, PET Club, ecc. (una prece, sigh!) - un carattere di maggior scavo in profondità sui grandi temi: quelli dell'informatica, degli algoritmi, delle emergenti problematiche d'utente. Il tutto a fianco della segnalazione dei prodotti; dei Bitest; delle realizzazioni hardware (la cui sezione rimane) della recensione dei package con il loro ruolo sempre più decisivo nella microinformatica.*
Si possono prevedere delle connessioni, a volte, tra le due anime del nuovo **Bit***: la parte generale-introduttiva magari teorica potrebbe andare su* **Bit***, diciamo così, vero-e-proprio, mentre* Riservato Personal *ospiterà, nel medesimo numero, la parte implementativa ed il listato. Facciano bene attenzione a questo criterio quanti si candidano come autori. Costoro poi tengano a mente che su* **Bit RP** *(d'ora in poi lo si potrebbe anche sintetizzare così) gli articoli dovranno avere un carattere di sintetica stringatezza (quindi bando alle sbrodolature umoristiche, salvo quelle fini e di buon gusto, ed in via eccezionale) però ciò non li esime dall'obbligo morale della buona documentazione: flow o diagrammi di struttura, continuo a raccomandare, voce di chi chiama nel deserto dei micropatiti interattivi ma tetragoni a simili quisquilie.*
Last but not least, Paolo Capobussi - finalmente coinvolto nei ranghi nostri in modo organico - sarà il curatore principale di **Bit RP***.*
E adesso, udite udite, abbiamo ampliato la nostra dotazione di personal fino ad includere tutti i tipi più diffusi, sui quali il Paolo, a chiunque bussi, risponderà scuotendo in senso affermativo il capo alla tassativa condizione che il programma (su cassetta come minimo!) giri a dovere.

P.S. - Dimenticavamo: incellofanato al presente numero si sarà potuto notare un corposo opuscoletto. Non è il solito gadget tra il frivolo ed il superfluo, zeppo di cifre che interessano al più i pettegoli del Villaggio Telematico prossimo-venturo, magari aggiornate al 1975 dopocristo. E nemmeno l'introduzione al funzionamento del ciclo FOR...NEXT con tutti gli arcani annessi e connessi. È un vero e proprio manuale professionale del più diffuso sistema operativo del microcomputer, quel CP/M giustamente battezzato "Software bus" per l'enorme contributo da esso dato alla diffusione dei programmi nell'area personal. Insomma una pietra miliare nella conoscenza di base di chi al personal computing voglia introdursi in modo serio, anche se agevolato (però senza mistificazioni pseudo-didattiche!).
Che dire di più? Null'altro che questo: non ci si indurre alla tossico-dipendenza da regalo-pesante. Perlomeno nel senso che ciò dipenderà molto dal favore massiccio con cui l'editore si augura che sforzi come questo vengano accolti.

Gianni Giaccaglini

Dita incrociate

Secondo gli osservatori, il settore dei piccoli elaboratori gestionali (dai 15 agli ottanta milioni di lire), registrerà incrementi annui non inferiori al 20%. Questo trend favorevole della domanda pare sia dovuto alle diminuzioni di prezzi delle unità centrali e dei dispositivi di memoria. Comunque, un fattore rilevante è la disponibilità di buoni pacchetti soft orientati al lavoro di ufficio e, non ultimo, il bell'aspetto estetico delle macchine. (Da una nota informativa di Bit Relations).

L'originale in copia

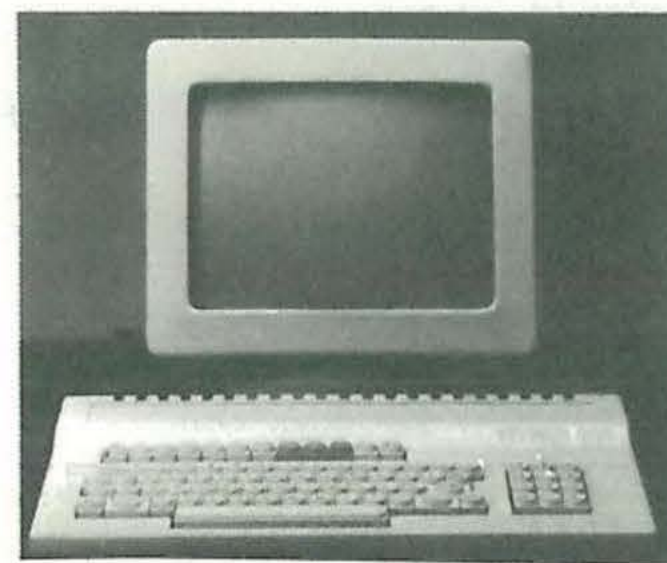
La Rank Xerox ha recentemente proposto a Londra con la linea "10" il suo concetto delle nuove copiatrici su carta comune. Tra i loro requisiti vi è che sono più affidabili, copiano sul retro del foglio, permettono di trattare anche originali come libri senza perdere troppo tempo, e si basano su tecnologie e elettronica assai sofisticata. Una famiglia di fratellini, verrebbe di pensare, sempre più concorrenti anche verso le stampanti come periferiche per personal computer

Allarme

La Senamion, una società affiliata alla Selesta Ingegneria, ha realizzato una serie di prodotti intelligenti per il controllo degli accessi e la rilevazione delle presenze, caratterizzati da una grande semplicità d'uso e dalla chiarezza dei messaggi con cui il sistema interagisce con l'operatore.

Selesta Ingegneria S.p.a
Via Meravigli, 16
20123 Milano
Tel. 02/802392

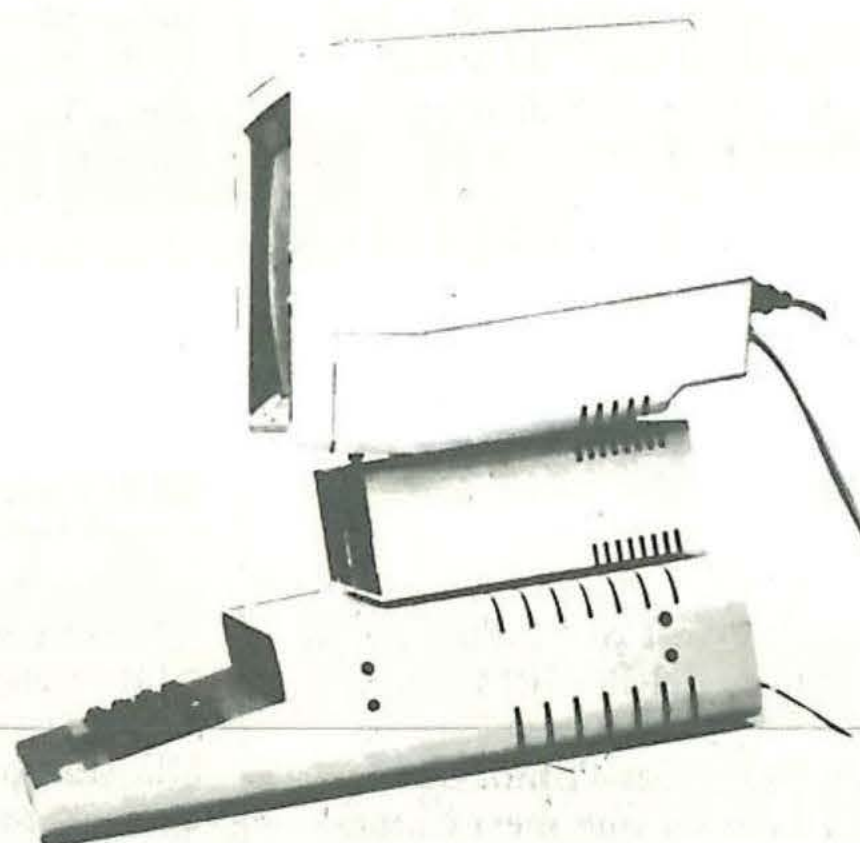
Terminale sdoppiato

La SPH di Milano ha reso disponibile un terminale in due versioni dedicate a due diverse fasce di utenza. I terminali si chiamano Viterm 520 e Viterm 520 Plus, dove il secondo è la versione più sofisticata del primo. Entrambi sono compatibili con VT52 Digital e Z19 Zenith. Oltre al set ASCII il Plus dispone di un set grafico di 33 caratteri. Il video è a fosfori verdi con banda passante di 30 Megahertz, e mostra 24 righe su 80 colonne. Le capacità di editing comprendono il controllo del cursore, l'inserzione di carattere o di linea, e la definizione di campi protetti o reverse video. Questa serie di terminali è dotata di una o due pagine di memoria, a seconda del modello, interfaccia seriale RS 232C con baud rate programmabile fino a 19.200.
La tastiera, a basso profilo, è dotata di 83 tasti e comprende un tastierino numerico.
La distribuzione dei prodotti della SPH è curata dalla Sitel in base a un recente accordo.

Sitel
Via Galvani, 2
20054 Nova M. (MI)
Tel. 0362/43008

Cresce il Personal Computer IBM

Con l'annuncio della disponibilità ufficiale anche per l'Italia del Personal della realcasa non è mancato l'annuncio di alcuni miglioramenti messi a disposizione degli utenti. Sostanzialmente si tratta di un miglioramento del sistema operativo DOS, di una accresciuta possibilità di connessione di periferiche (fino a tredici) e della possibilità di dotare il PC di uno o due dischi rigidi da 10 Mbyte ciascuno. Viene inoltre annunciato un nuovo modello di personal computer IBM, denominato XT, che comprende in configurazione base oltre alla tastiera, al video e alla stampante, una unità centrale con memoria utente di 131 Kbyte, una unità a disco flessibile da 5 pollici da 368 Kbyte, una unità a disco fisso da 10 milioni di caratteri, otto attacchi per unità periferiche e un adattatore asincrono per comunicazioni.

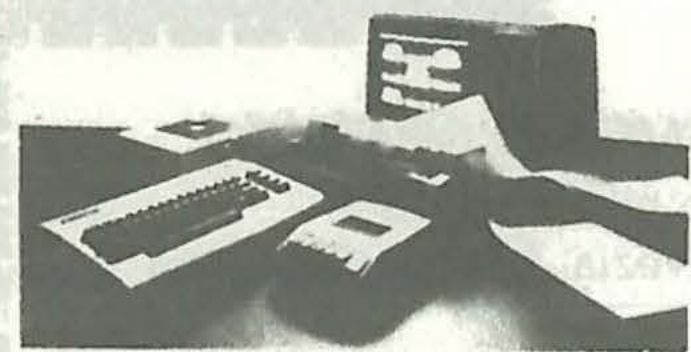
IBM Italia
20090 Segrate (MI)
Tel. 02/75484550

ELEDRA
PERSONAL COMPUTER
NEWS
FEBBRAIO 1983
12
PUBBLICAZIONE GRATUITA
GRUPPO ELEDRA
LEMON II
Personal Computer
PROGRAMMI COMPATIBILI APPLE II
PRODOTTO NAZIONALE
PREZZO COMPETITIVO
ELEDRA 3S S.p.A. - Viale Elvezia, 18 - 20154 Milano
IN VENDITA PRESSO I RIVENDITORI AUTORIZZATI PERSONAL COMPUTER ELEDRA 3S
GIUGNO 1982
1
RICHIESTA DI ABBONAMENTO GRATUITO
Spedire il coupon in busta chiusa a:
ELEDRA 3S S.p.A, - Viale Elvezia, 18 - 20154 Milano
BIT 12
☐ Desidero ricevere regolarmente Eledra Personal Computer News
☐ Ricevo già EPCN. ☐ Desidero avere informazioni su LEMON II
☐ Indicatemi il vostro rivenditore piú vicino.
Cognome e nome
Tit.
Attività
Ditta
Indirizzo
CAP
Città
Tel.

Novità dalla Zelco

La Zelco di Milano ha reso nota la disponibilità di un convertitore parallelo/seriale (RS 232 o current loop), che permette di trasformare l'uscita Centronics o compatibile, in seriale. Il convertitore, destinato ai distributori, permette di tenere a stock stampanti con un solo tipo di interfaccia e inoltre sostituisce il cavo di collegamento della stampante. Il convertitore comprende oltre al cavo una scatoletta, grande come un pacchetto di sigarette, che contiene un disaccoppiatore ottico e che trae l'alimentazione dalla stampante. La velocità di trasmissione seriale è selezionabile e può giungere a 9.600 Baud.
Viene inoltre annunciata la disponibilità di un pacchetto di word processing per la stampante Centronics 351 che ne utilizza tutte le possibilità, sia di alta definizione del carattere che di spaziatura proporzionale.

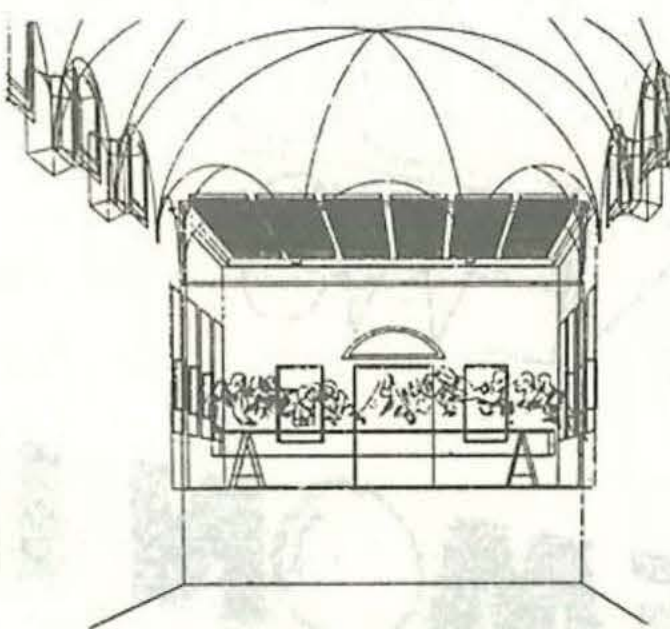
Zelco S.r.l.
Via V. Monti, 21
20123 Milano
Tel. 02/803336

Nuovo programma per l'economia casalinga con il VIC 20

La crescente diffusione del VIC 20 della Commodore ha indotto la produzione di software anche su questa fascia di home computer a basso costo.
"Contabilità giornaliera" è il nome di un package prodotto dalla Commodore, residente su cassetta magnetica, il quale può gestire le entrate e le uscite dell'amministrazione familiare.

L'amministrazione del menage familiare è sotto controllo e registrata in archivio. Si possono fare previsioni di spesa, bilanci e verifiche. Il programma, che comprende 5 conti di entrate e uscite ciascuno con 9 sottoconti per voci diverse, ha un prezzo molto interessante, 42.000 lire più IVA.

Commodore italiana S.r.l.
Via F.lli Gracchi, 48
20092 Cinisello B.(MI)
Tel. 02/6125651

Anatomia di un dipinto

Comprendere come pensava e vedeva l'artista che ha dipinto qualche opera famosa potrebbe sembrare una operazione puramente intellettuale. La regione Lombardia in collaborazione con la Computervision, ha mostrato che l'operazione era invece realizzabile in pratica. Il soggetto scelto per tale dimostrazione è il famoso dipinto di Leonardo da Vinci che si trova nel refettorio dell'ex convento di Santa Maria delle Grazie a Milano, intitolato "L'ultima Cena".
Parametrizzando ogni punto del dipinto e stabilendone "la profondità" in accordo con le leggi della prospettiva, si può ricostruire il gruppo scenico nella

memoria di un calcolatore per CAD (Computer Aided Design) e poi proiettare l'immagine su uno schermo. La si può quindi ruotare, osservare da dietro, da lato e sopra. Si entra cioè nella scena come se questa fosse a tre dimensioni. Peccato che dall'interno della scena non si possa fare altrettanto per ricostruire il mondo esterno.

Quattro pacchetti professionali per il PC IBM

La Hard & Soft di Rimini, primo importatore del PC IBM, rende disponibili quattro software di ottimo livello per l'unità della realcasa che da due mesi è stata introdotta in Italia anche ufficialmente. Tre sono di indirizzo contabile: gestione condomini, gestione contabile/amministrativa, contabilità semplificata; l'ultimo è uno strumento per la progettazione di strutture spaziali in zona sismica.
La gestione amministrativa è dedicata alle aziende e comprendente contabilità generale, fatturazione, magazzino, oltre a varie utilità come l'aggancio automatico tra queste varie procedure. Essa viene realizzata nella versione di base su dischetti da 160 Kbyte, ma può venire potenziata se si scelgono dischetti da 320 Kbyte o un disco rigido.
La contabilità semplificata permette le normali operazioni giornaliere, quelle trimestrali (IVA, deleghe bancarie ecc.) e quelle anuali (IVA annuale, dichiarazione dei redditi ecc.). Il programma di gestione condomini prevede 9 procedure principali (gestione archivi, stampa piani dei conti e elenco fornitori) e 11 sottoprocedure che giungono sino alla stampa delle convocazioni per una assemblea. Il calcolo delle strutture in zona sismica si basa sull'analisi statica dei carichi sui vari elementi della struttura in cemento armato, e sulla ripartizione delle sollecitazioni dovute al movimento sismico sui vari elementi di controventatura della struttura (questi controventi vengono forniti all'inizio come matrici generiche modificabili). I telai vengono risolti con il metodo delle deformazioni. Anche in questo programma vi sono alcune procedure di base (otto) e di quattro sottoprocedure.

Hard & Soft Systems S.r.l.
Via Valturio, 43
47037 Rimini
Tel. 0541/773343

Fino a quattro Apple su disco Bernoulli

La Informatica Computer Center di Modena ha realizzato un sottosistema che permette di gestire una unità a dischi del tipo Sparrow 10 da 10 Mbyte, fornita dalla General Automation, mediante un Apple II. Il sottosistema che realizza l'interfacciamento include un multiplexer che permette di collegare al medesimo disco fino a quattro computer. Assieme all'hardware vengono forniti anche alcune utilità software residenti su un disco in cartuccia e studiate per permettere la copiatura traccia a traccia (per il backup), la gestione di più dischi e di file molto lunghi. Per questo sistema sono stati anche sviluppati dei programmi applicativi che tendono a rendere l'apparecchiatura anche più appetibile.

General Automation Italia S.p.a.
Via Gasparre Gozzi, 1/A
20129 Milano
Tel. 02/718531

Imparare giocando e nuovo home dalla Texas

La Texas Instruments ha annunciato la prossima disponibilità di undici nuove cartucce di software allo stato solido per l'home computer TI99/4A. La prerogativa di queste cartucce è che fanno esercitare chi le usa nella pratica delle operazioni aritmetiche inserendo dei quiz all'interno di giochi molto simili ai videogiochi più noti.
A proposito del TI 99/4A vengono definitivamente rassicurati i venditori e gli utenti che i trasformatori di questi computer sono completamente affidabili e non presentano rischio alcuno per l'operatore. Questa precisazione si è resa necessaria dopo che negli Stati Uniti si era riscontrato in prove di laboratorio che il trasformatore a 110 V poteva dare dei problemi di sicurezza.
Contemporaneamente viene annunciato il Computer BASIC TI99/2. Questo sarà un'unità di prezzo veramente ridotto, costando circa 180.000 lire, con la possibilità di usare sia cassette che cartucce allo stato solido. L'unità può disporre di comunicazione tramite interfaccia RS 232, o di periferiche quali il registratore per cassette e il dispositivo di stampa o il plotter. Assieme al TI 99/2 viene fornito un adattatore per corrente alternata e un modulatore per RF che ne permette l'uso anche con un televisore domestico.
La disponibilità del nuovo computer in Europa è prevista per il terzo trimestre di quest'anno.

Texas Instruments Semiconduttori Italia S.p.a.
Viale Europa, 38/44
20093 Cologno Monzese (MI)
Tel. 02/2532451

Stampanti Honeywell HISI in Cina

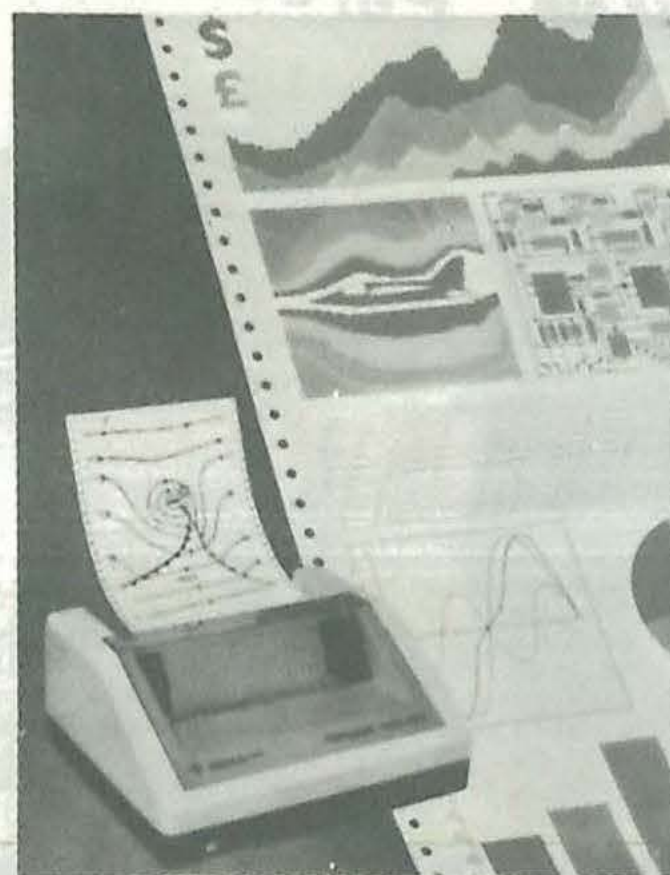
Dopo aver venduto un primo contingente di stampanti seriali a matrice del modello S31 ad alcune università cinesi, la HISI spera di poter entrare in modo più sostanzioso in questo mercato ancora intoccato per farvi buoni affari. Nel frattempo anche le attività "domestiche" della HISI stanno allargandosi con la designazione di un distributore per le stampanti seriali a matrice in Italia con la Data Base OEM di Vimercate e in Francia con altre due società che curano la parte settentrionale e meridionale del paese nostro vicino.

Un leoncino in frac

Una novità dalla americana Cromenco riguarda l'annuncio di un personal computer denominato C 10. L'unità è stata dotata di memoria centrale di 64 Kbyte di RAM e di altri 24 Kbyte memoria di sola lettura o ROM. Il microprocessore scelto è lo Zilog Z80A, operante alla frequenza di 4 MHz. La capacità della memoria di massa su dischi flessibili da 5", è di 390 Kbyte. Nella configurazione di base viene fornito un solo drive, ma se ne possono aggiungere altri tre.
La tastiera è separata dal video e appare abbastanza maneggevole. Sia la scheda madre che le interfacce (RS 232 e Centronics), come l'elettronica di gestione del video, sono tutte contenute nel mobiletto a fianco del cannone elettronico. Questo può essere montato su un supporto che ne permette l'orientamento ed ha un formato di visualizzazione di 24 righe orizzontali per 80 colonne verticali su uno schermo di 12" a fosfori verdi. Il C 10 ha la compatibilità verso elaboratori IBM con i quali può operare come terminale intelligente.
Il sistema operativo di cui l'unità è dotata è compatibile con il CP/M, di cui presumibilmente rappresenta una versione migliorata. I linguaggi di programmazione attualmente disponibili sono: BASIC, COBOL, Assembler, FORTRAN IV, Ratfor, Macro Assembler e Lisp. E' interessante notare che, nonostante la notevole configurazione di base, il prezzo cui questa unità è stata posta sul mercato si mantiene inferiore ai 4 milioni di lire, che comprendono un programma per la pianificazione finanziaria VisiCalc-like e uno di elaborazione di testi, oltre al C-DOS (il sistema operativo) e al BASIC. Pur nella sua ridotta dimensione, si ha l'impressione che la macchina sia un leoncello, come dire che abbia grinta da vendere. Questa unità viene importata e distribuita in Italia dalla FB S.r.l. di Chiesina Uzzanese, che sta cercando dei distributori, anche esclusivi, per varie aree geografiche della penisola.

FB S.r.l.
Via privata Delle Rose, 11
51013 Chiesina Uzzanese (PT)
Tel. 0572/48861

Nuova stampante a colori della Gould presso la Perimel

Una interessante novità, recentemente introdotta dalla Gould Brians, è costituita da un plotter a sette penne, completamente miscelabili, denominato "Printagraph Multicolor".
La macchina è particolarmente indicata per produrre stampe con l'uso di caratteri grafici (tutto il

set PET graphic) e per la tracciatura di schizzi e diagrammi, sia nel CAD che nel controllo di processo. Inoltre la definizione del set ASCII entro una matrice di 6x7, può interessare in applicazioni per produzione di testi scritti. Il prezzo per l'unità con il formato A4 si aggira con una interfaccia, parallela oppure seriale, attorno ai 6.7 milioni.

Perimel S.r.l.
Via Fezzan, 9
20146 Milano
Tel. 02/4225945

È nato il Sinclub

Il Sinclub è un superclub: non vuole cioè, sostituirsi ai Club già esistenti, nè a quelli nasciturì. Al contrario: vuole che di Club ce ne siano sempre di più, e agisce in modo che ciò si avveri, favorendo gli incontri e promuovendo l'associazionismo fra utilizzatori dell'ormai vasto sistema ZX.
Il Sinclub desidera che i Sinclair Club siano attivissimi e ne inventino una più del diavolo: la fantasia dei Sinclairisti, si sa, è inesauribile.
Il Sinclub, fa circolare le idee, le proposte, i bollettini, e tutto quel "prodotto della cultura Sinclair" che ormai ha un ruolo determinante nell'evoluzione della microinformatica in Italia e nel mondo.
La sede centrale del Sinclub è situata presso la redazione della rivista Sperimentare, che funziona da connettore fra tutti i Sinclair Club presenti e futuri: raccoglie informazioni, richieste di contatti, dati sulla consistenza del "parco Sinclair" in Italia, e, soprattutto, offre a tutti i Sinclair Club un esclusivo servizio gratuito di certificazione del software originale. Ciò significa che i programmi prodotti da un Sinclair Club saranno valutati e certificati nella loro validità dal Sinclub e quindi catalogati in un bollettino centrale, in modo che tutti gli altri Sinclair Club siano in grado di apprenderne l'esistenza e, se ne han bisogno, di richiederli per utilizzarli.
Le modalità di fruizione di questo servizio non sono ancora stabilite, ma è certo che chi fornirà alla banca di software del Sinclub programmi validi, riceverà in compenso un buono di importo proporzionale alla validità del programma inviato, buono che potrà utilizzare a sua volta per acquisire altri programmi elaborati da altri Sinclair Club.
È nella sede centrale che si dà luce all'organo ufficiale, cioè al Sinclub Sperimentare pubblicato sulla rivista.
Inutile dire che il Sinclub attende con ansia interventi dei Sinclair Club, che saranno pubblicati con tanto di citazione degli autori e dei Sinclair Club di appartenenza se giudicati interessanti.
Ed è sempre la sede centrale del Sinclub a mantenere i necessari collegamenti con la stampa italiana ed estera per informare ed essere informati. Anche i contatti con organizzazioni similari estere sono curate dal Sinclub. Nel campo dell'animazione, è logico che la sede centrale divulgherà e diffonderà tutte le iniziative dei singoli Club di cui sarà a conoscenza.
Ma ancor di più: non appena terminato il censimento dei Sinclair Club il Sinclub indirà gare sia fra Sinclair Club sia fra singoli soci, in modo da creare quella giusta e sana competizione sportiva che tiene tutti su di giri, concorrenti e spettatori.
Inoltre ai vari Sinclair Club saranno attribuiti punteggi in tema di efficienza, abilità programmatoria, animazione e attivismo, in modo da premiare i Sinclair Club che si dimostrano più attivi. Naturalmente si terrà conto delle realtà locali più o meno penalizzanti.
Il tutto sempre in un'atmosfera sportiva e amichevole!
Per aderire al Sinclub e se avete un Sinclair Club, uno Zx Club, o come altro lo avete chiamato, non c'è nessun bisogno che vi affrettiate a cambiargli il nome in Sinclub: al contrario, ciò farebbe un po' di confusione e basta.
La cosa invece utile da fare è scrivere alla sede centrale, possibilmente utilizzando il tagliando Censimento Sinclub che trovate su questo stesso numero di Bit.
Dovrete indicare chiaramente chi è il Capoclub: non per essere gerarchici o militaristi, ma dobbiamo sapere a chi fare riferimento, con la corrispondenza, per l'invio di materiale omaggio o della stampa sociale.
Il Capoclub, dunque, riceverà tutto ciò che è necessario: regolamento, stampa, tessere di adesione. Insomma tutto.

Il primo personal computer portatile è stato ideato, con ogni probabilità da Adam Osborne, noto consulente del settore e technical writer di successo. Questo prodotto ha preso il nome del suo ideatore: Osborne 1.

Forse il più affascinante portatile oggi sul mercato: l'HX20 Epson. La tastiera è tipo macchina da scrivere, a 68 tasti, mentre il display è su 4 linee da 20 caratteri, LCD. L'unità centrale è costituita da due microprocessori realizzati dalla giapponese NEC.

Il New Brain, della Grundy Business Systems Ltd. Si noti la tastiera QWERTY a 62 tasti, minuscole e maiuscole. Il display è su una linea da 16 caratteri, ciascuno su 14 segmenti.

di A. Cavalcoli

Introduzione

Una nuova famiglia di PC inizia sempre più ad affermarsi sul mercato. Si tratta dei cosiddetti "portatili", caratterizzati da una compattezza molto spinta, quindi dimensioni ridotte, ampie possibilità di estensione, prestazioni interessanti, comparabili a quelle degli usuali personal. Questa nuova famiglia mette ancora una volta di più in crisi la dizione omnicomprensiva di personal computer, la quale a questo punto necessita di una più precisa articolazione, sia in termini di potenza, che di configurazione, e di conseguenza, di ambiente d'uso, per non dire "modalità" d'uso.

Aggiornamento di una definizione

All'inizio c'era il computer o calcolatore o cervellone. E in effetti c'è ancora.
Anzi, in alcuni casi c'è solo "lui", il cervellone.
Si legge su La Repubblica di sabato 20 Novembre 1982: "Firenze spinge un tasto e il 13 esce dal cervellone".
Evidentemente si tratta di un sistema installato in un bar della città, ad uso dei clienti che desiderano compilarsi elettronicamente la schedina, associando al piacere ed alla passione del Totocalcio quel certo "frisson" futuristico che sta facendo il successo dell'iniziativa.
Scorrendo l'articolo a pagina 17, si legge poi: "Il cervellone, chiamato Apple II, ...".
Ma allora, come la mettiamo? Non si era detto (voi giornalisti) che il Cervellone, con la "C" maiuscola, era quello che sputava le pensioni, quello che dava i risultati elettorali, quello che ci scheda nell'anagrafe tributaria? Diventa allora cervellone pure il più classico dei PC, l'Apple? Va da sé che, se la stampa fa opinione e cultura, siamo d'ora in poi abilitati a dire: "Senti, passami per favore il cervellone, che devo fare questa radice quadrata" (con ciò riferendosi ad una innocua calcolatrice tascabile).
Oppure: (al centro di calcolo del ministero delle finanze) "Bagazzi, accenda un po' il cervellone, che sputiamo fuori un po' di cartelle delle tasse".
Il marasma è completo, e le masse reclamano cultura e punti di riferimento sicuri: quindi, approfittando di questo articolo sui portatili, due parole di inquadramento, valide oggi, forse non più domani, a seguito di altre modificazioni tecnologiche o solo di nuove configurazioni di prodotto. Il livello più alto è rappresentato dai large computer, cui si attribuisce senza pecca anche la direzione di mainframe, con ciò intendendo grossi sistemi, installati in locali appositi, ad aria condizionata, con più addetti occupati alle varie possibili operazioni, comunque di ampia portata gestionale.
Alla fine, IBM e simili.
Scendendo, incontriamo i minicomputer, più familiarmente detti mini. Più lenti dei mainframe, vedono applicazioni sia gestionali che di controllo di processo.
L'unità centrale è usualmente una logica elettronica ad alta velocità, realizzata in "discreto", cioè più componenti specializzati che nel loro insieme danno la CPU.
Poi ecco i microcomputer, computer aventi come unità centrale dei microprocessori: qui inizia il difficile.
La precedente definizione inizia a sapere di stantio, sia per l'attuarsi di ampie fascie di sovrapposizione con i mini, sia per l'accresciuta potenza dei microprocessori, soprattutto i 16 ed i 32 bit, per potenza tranquillamente paragonabili alle CPU discrete dei mini.
Allora posso fare un mini usando come CPU un 16 bit "super" come lo Z8000: ma non ottengo forse un microcomputer?
È un problema, risolvibile solo se vengono individuati dei parametri caratteristici che separino l'oggetto dagli altri, senza ambiguità.
Ad esempio si potrebbe utilizzare la potenza elaborativa, la velocità, il costo.
Ma non, comunque l'applicazione, perché nessuno vieta che io utilizzi un mini DEC per

Una nuova famiglia: i personal portatili

gestire un motore passo-passo.
Volendo essere più seri, la parametrizzazione dell'applicazione è sconsigliata dai diversi livelli di sofisticazione di uno stesso tipo di applicazione, essendovi processi industriali controllabili da microcomputer general purpose specializzati dal software, da sistemi dedicati quali i controllori programmabili, oppure da mini veri e propri.
Allora, prendendo velocità e costo (il quale è somma evidentemente di molti fattori: affidabilità, qualità, periferiche, software, assistenza, potenza, ecc.), già si riesce a fare un primo ordine nella massa pulsante dei cervelloni.
Quindi eccoci ai microcomputer. Subito però siamo assaliti da altre etichette: personal, hobby, home, e, ultimo per ora, portatile.
Diciamo subito che portatile non ha nulla a che fare con pocket (computer), anche se un pocket, stando in tasca, è evidentemente portatile. Penso che ci si intenda.
Allora, stando nei microcomputer, così definiti dalla presenza a livello di CPU di un microprocessore, perché personal, home, hobby? (Tanto per iniziare).
E ancora, che senso ha sentire parlare di small business systems, od altre dizioni analoghe che tendono a caratterizzare il futuro operativo di quello che a prima vista appare essere un personal?
Con il rischio di essere noioso, vorrei agganciarli un po' alla storia. Ritengo che la dizione personal, decisa in USA all'inizio degli anni '70, abbia più che altro un significato psicologico: finalmente ecco un elaboratore che esce dagli asettici centri di calcolo, che posso fare mio ad una spesa ragionevole, dell'ordine di grandezza (classico paragone d'altri tempi) di un buon impianto Hi-Fi.
Notate che sempre e comunque di microcomputer si tratta, stante la CPU microprocessore: ma per microcomputer si intende poi tutta una serie non inquadrabile di "oggetti elettronici" adibiti a mansioni di controllo tra le più varie, di orientamento industriale, il più delle volte di definizione custom a seconda della situazione.
Allora personal indica un sistema general purpose, da adibirsi ad usi "personalmente" decisi, di sofisticazione legata alla potenzialità della macchina.
Ne consegue una possibile accettazione delle altre dizioni: con hobby si indica un microcomputer general purpose, cioè personal, con il suo corredo minimo di funzioni elaborative e di periferiche od interfacciamenti previsti, che per potenza difficilmente esce dal contesto hobbistico, legato al piacere dell'uso, ai giochi, alle applicazioni sperimentali autonomamente decise.
Ritengo sia chiaro che il costruttore ha pensato di dare un corretto corredo alla macchina, che svolge "bene" le funzioni hobbistiche. Più critica la definizione di home computer, che vediamo tranquillamente attribuire al personal della TI.
In questi casi direi che si tende ad enfatizzare il futuro ambiente d'uso, quindi la casa, con il supposto soddisfacimento delle esigenze applicative presenti e future dei vari membri costituenti.
Il padre si fa la contabilità, la gestione della mailing list, della biblioteca, la madre tiene i conti di tutti i giorni, magari realizza un calendario personale con evidenziati i compleanni e gli onomastici di tutto il parentado, il figlio impara il BASIC, sperimenta un po' di CAI, gioca come un matto.
E tutti alla sera, nonno compreso, ci danno dentro con Invaders e similari. Alla fine l'ambiente è dato dal corredo, che suggerisce l'uso, oltre che stimolarlo con una adeguata politica dei prezzi.
A questo punto viene da sé una giustificazione dei cosiddetti small business computer, evidentemente corredati in modo opportuno (hardware più software) e di adeguata potenza elaborativa, per soddisfare esigenze contabili di varia natura.
E il portatile? Ritengo non sia da inquadrarsi nel modo fin qui seguito, ma piuttosto da vedersi "in parallelo", in quanto questa sua portabilità si somma ad altre caratteristiche d'uso, originando l'hobby portatile, lo small business portatile, e così via.
Portabilità come ulteriore caratteristica, analoga ad esempio, alla gestione del colore, che prima non tutti avevano, e che ora c'è in modo differenziato, da cui una maggiore o minore apprezzabilità da parte del pubblico.
Portabilità come elemento commerciale, per colpire ambienti poco interessati finora.
Comunque, grosso passo tecnologico, in quanto oggi un oggetto personal computer portatile, non perde, per la sua portabilità, prestazioni o capacità elaborative rispetto a molti classici personal "inamovibili".

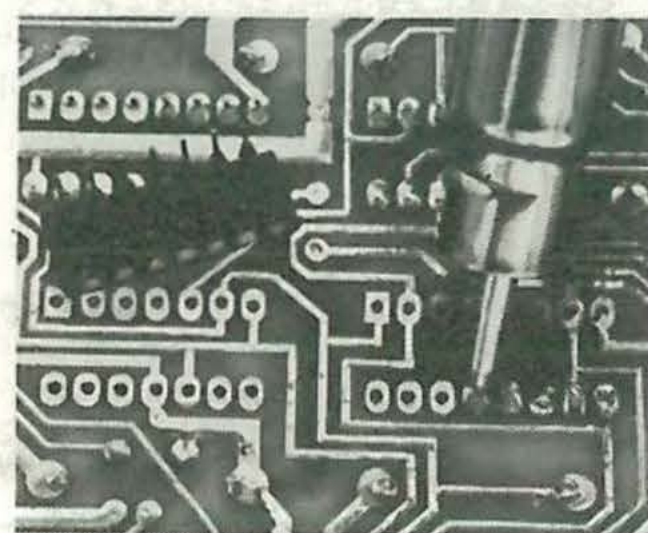
ZX Spectrum Sinclair. Questo personal si presenta, come del resto i suoi predecessori Sinclair, in forma compatta: un contenitore di dimensioni circa doppie rispetto una normale calcolatrice tascabile. Il corpo contenitore comprende la tastiera a 40 tasti e la scheda unità centrale.

I personal computer portatili

Il fenomeno dei PC portatili è da

inquadrarsi in una sempre più vasta azione dei costruttori, che dal prodotto di base PC iniziano a creare, a parità di prestazioni, oggetti fisicamente differenziabili, con l'obiettivo di coprire esigenze e comodità d'uso che possono dar luogo ad un allargamento del mercato.
Ma non solo questo.
Con i portatili viene stimolato l'utente, gli si offre un prodotto senz'altro psicologicamente appagante e motivante, riconducendo nella sfera PC attività e momenti operativi, esistenziali, che, senza i portatili, mal vedrebbero un PC quale elemento risolutore.
Banalmente parlando, in viaggio non ci si portava un PC mentre oggi è possibile.
Ancora, la miniaturizzazione dei PC conferisce a questi prodotti una "anima" diversa.
Prendete l'orologio, gli occhiali, la radiolina, la penna, l'accendino, insomma tutti gli oggetti personali che a seconda delle nostre esigenze sia personali che del momento, ci accompagnano.
Fanno oramai parte dell'abito che indossiamo: il *mio* accendino, la *mia* calcolatrice, il *mio* registratore portatile.
Cosa di più personale, privato? Questi oggetti possono essere anche visti come una estensione del proprio essere, delle proprie capacità, in termini di efficienza operativa immediata.
È sempre più usuale vedere sul treno il vicino con la cuffia del Walkman, vuoi per distrarsi, ascoltando musica, vuoi per ripassare una lezione di inglese, approfittando del "vuoto di attività" dato dal viaggio in treno.
Per non parlare poi di quanti, realizzando in pieno il nome di questo oggetto, lo utilizzano passeggiando, rinchiusi nel loro mondo musicale. Allora, perché non accettare la visione di un PC sulle ginocchia del professionista in viaggio, oppure, la sera, nella camera d'albergo, per verifiche, calcoli, impostazioni?
Certo, l'ideale è il super-gadget da polso: lo si accosta alla bocca e, vocalmente, si pongono i quesiti più vari; lui, in costante contatto con il Cervellone Totale, risponde in modo completo e personale.
Questa è, per ora fantascienza, neppure i giapponesi ancora ci sono riusciti.
Per ora, accontentiamoci dei PC portatili, forniti di dati ed informazioni, vera informatica distribuita, anzi direi "viaggiante".

Quali e quanti

Il capostipite della famiglia dei portatili è stato l'Osborne 1, ideato da Mr. Osborne, noto consulente ed ora imprenditore del settore.
Sinceramente io, come molti altri, guardammo con un certo scetticismo questo prodotto, con la tendenza a definirlo più una curiosità che non un sistema di sicuro avvenire.
Ed ammetto anche che mi era sfuggita la carica rivoluzionaria della sua portabilità.
Ma penso di essere scusato dall'inevitabile costante provincialismo contro il quale tutti noi quotidianamente combattiamo, vivendo in un paese che non è l'America, in cui tutto succede, ma che sente in modo disomogeneo effetti e prodotti, comunque in seconda battuta, non sulla scorta di esigenze che altrove motivano l'uscita di prodotti. Qui, il più delle volte, ai nuovi prodotti importati noi ci adeguiamo, inventandocene le ragioni d'uso successivamente.
Ma torniamo ai portatili; quali e quanti?
Obiettivo di questo articolo è quello di realizzare non certo una ma una informativa di base, soprattutto rivolta a prodotti abbastanza nuovi, giusto per permettere un "feeling" del trend di sviluppo. Non parlerò quindi

È morto, a vent'anni, ***Nicola Ricci****, uno dei primi collaboratori di Atari Club.*
Ci eravamo conosciuti parecchi anni fa, ad una qualche fiera, lui veniva la mattina con i suoi spartiti e passava tutta la giornata lì con il Music Composer, e mi piace ricordarlo così, mentre discutendo di DOS e di RS232 insegnava ad Atari Bach....

dell'Osborne 1, già altrove trattato ampiamente, né dell'HP-85, anch'esso ben noto, e per alcuni versi inquadrabile nell'ambito dei portatili.

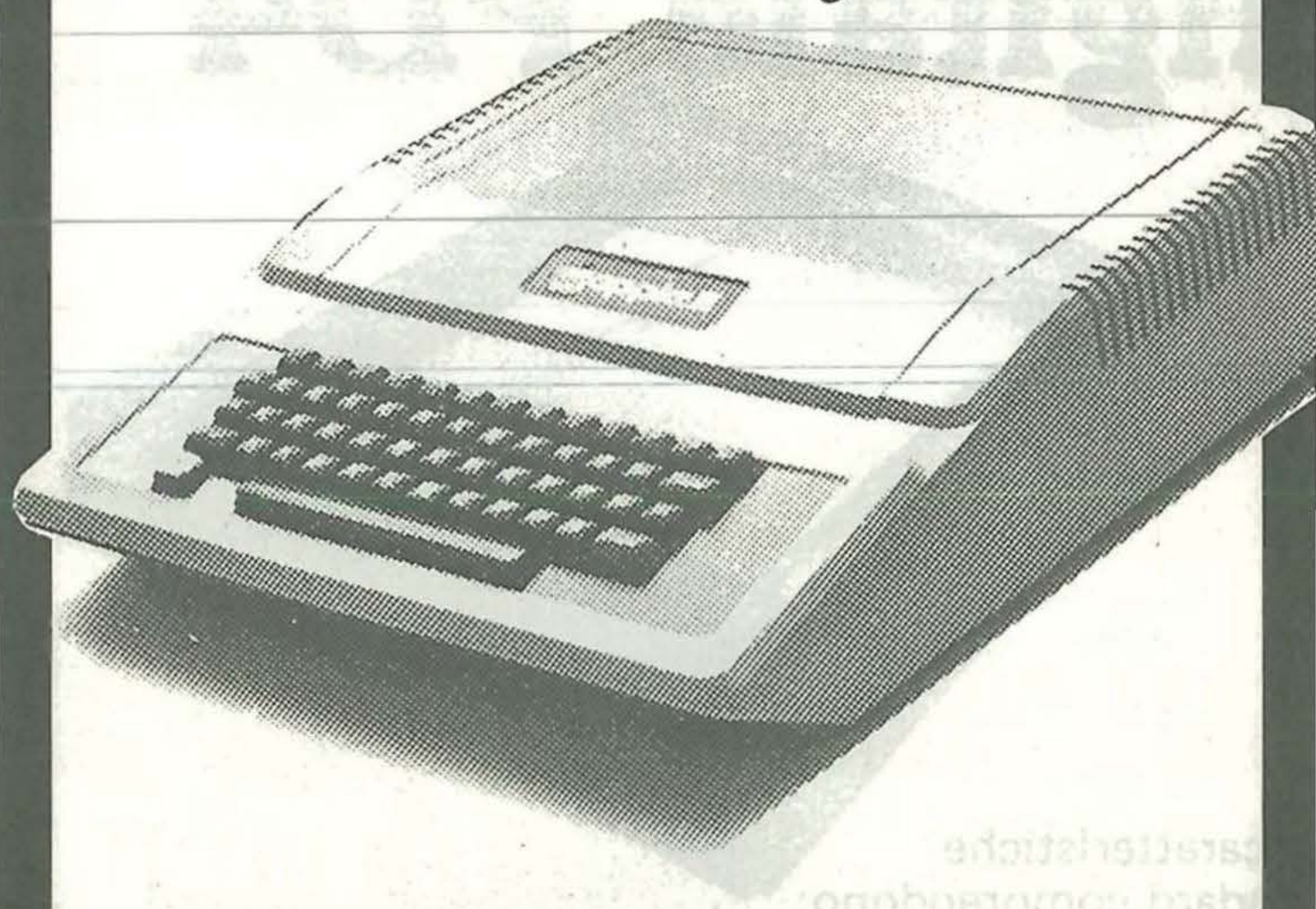
Il gioiello Epson

A mio giudizio questo PC è il più pregevole tra i portatili, sia per prestazioni che per eccellente equilibrio tra dimensioni e portabilità. Questo equilibrio è molto importante, perché posso avere un portatile che, chiuso nella sua custodia, è grande come una macchina da scrivere tipo le ultime elettroniche Olivetti, oppure come una macchina da cucire, ed un portatile che posso mettermi in tasca, dando subito adito a non poca confusione con i classici pocket.
Questi due estremi di dimensioni danno poi luogo a pratiche di lavoro più o meno comode: tasti piccoli, display limitato, poca memoria, oppure esigenza di ampio corredo di periferiche per realizzare una soddisfacente configurazione, da cui una negazione di fatto della portabilità, e così via.
Nel caso Epson, il loro portatile HX-20 pesa poco più di un chilogrammo e mezzo, con dimensioni 21 x 29 cm, spessore 4.5 cm, quindi del formato della presente pagina di **Bit**.
Le caratteristiche tecniche, viste le dimensioni, sono stupefacenti. La memoria RAM è CMOS, 16 Kbyte con estensione fino a 32 Kbyte. La ROM è su 24 Kbyte, ma può arrivare ai classici 64 Kbyte.
Su questa ROM si trova il BASIC, che è Microsoft esteso ed un sistema operativo originale Epson, quindi non compatibile con altri classici SO. L'unità centrale è rappresentata da due microprocessori ad 8 bit, realizzati custom dalla NEC per la Epson, ma pare, derivati dal 6800 Motorola.
I due micro agiscono secondo l'organizzazione master-slave. La tecnologia realizzativa dell'HX-20 è CMOS da cui la possibilità di alimentazione con batteria al cadmio-nickel, autonomia di 50 ore, ricaricabilità in 8 ore.
La visualizzazione avviene per display incorporato LCD, 4 linee da 20 caratteri, con grafica 120 x 32 punti, che diventano 128 x 96 se viene utilizzato un monitor esterno, interfacciabile.
Dato che i cristalli liquidi non emettono luce, si dipende dalle condizioni ambientali per una buona lettura: per agevolare comunque l'utente, il display è orientabile meccanicamente, cosa molto utile. La capacità dello "schermo" incorporato è evidentemente molto bassa, per cui, da una prima impressione, deludente.
Ma questo portatile, come altri, agiscono secondo la nota tecnica dello "schermo virtuale": infatti le 4 linee da 20 caratteri ciascuna sono solo una parte dello schermo completo (virtuale), presente in memoria, di 255 linee.
Quindi sul display LCD incorporato è realizzata una finestra, spostabile sia orizzontalmente che verticalmente, per cogliere la parte di interesse.
Infine la tastiera. Questa è su 68 tasti, del tipo QWERTY, dove con questa definizione si indicano le posizioni dei primi 6 tasti alfanumerici in alto a sinstra (il primo tasto alfabetico è la lettura Q). Su una macchina da scrivere del tipo Olivetti lettera 32 (anch'essa portatile!) La designazione sarebbe QZERTY.

ZX Spectrum Sinclair

La Sinclair inglese è famosa per i suoi piccoli PC: ZX80 e ZX81. Ora arriva lo Spectrum, oggetto veramente compatto, 140 x 100 mm, praticamente il doppio di un normale pocket.
Il box è costituito dalla tastiera, QWERTY, di 40 tasti, con ben 191 funzioni, e da una scheda con tutta la sua logica:

- CPU microprocessore Z80 da 3,5 MHz (frequenza in effetti inusitata)
- 16 Kbyte ROM per BASIC e SO
- RAM da 16 Kbyte fino ad un massimo di 32 Kbyte
- Modulatore UHF per TV, B/N o colore, ma PAL
- Altoparlante miniaturizzato piezoelettrico (quello delle piccole sveglie, tanto per in-

tenderci)
- Interfaccia cassetta audio
- Interfaccia stampante, tipo ZX Sinclair

È in preparazione una unità a micro-dischi da 100 caratteri. Come ben sanno gli amatori ZX, i precedenti modelli 80 ed 81 possiedono un BASIC che in effetti è un dialetto, qui riproposto nello Spectrum anche se in versione più completa, con possibilità di definizione di funzione su una linea (*DEF FN*) e creazione e lettura tabelle di costanti (*READ, DATA, RESTORE*).
L'operatività generale è incrementata, a livello scrittura di programma, dal fatto che ogni tasto della tastiera è una lettera od un numero, ma anche una funzione, una istruzione BASIC. Esiste poi un interessante strumento di autoapprendimento, se nella scrittura di un programma vengono commessi errori di sintassi, questi sono scoperti a livello tasto ed esistono codici particolari atti a guidare l'utente verso la soluzione.
Molto valida, infine la grafica a colori: otto colori, con risoluzione grafica 256x192.
Il difetto che subito appare è la ridotta dimensione dei tasti, cosa che può creare problemi ad individui "robusti".
Ancora, il SO è particolare dello Spectrum, per cui si determina una critica limitazione nell'utilizzo di esistenti biblioteche di programmi.
Orientamento: applicazioni didattiche e giochi, mentre per l'ambito professionale solo situazioni di ridotta complessità possono essere approcciate.

New Brain

A conclusione di questa stringata panoramica di novità "portatili", un prodotto molto interessante dall'Inghilterra: il New Brain della Grundy Business Systems Ltd, società in effetti poco nota in Italia. Di ridotte dimensioni (256 x 15 x 45 mm), si presenta come un pocket con display su una linea da 16 caratteri alfanumerici.
La limitatezza del display viene ovviata da due accorgimenti tecnici. Prima di tutto, il display non è LCD, ma a LED su 14 segmenti (il doppio del normale), da cui la possibilità di visualizzare senza difficoltà e con ottimi risultati, tutti i caratteri alfanumerici.
Poi, anche nel New Brain, viene usata la tecnica della finestra su un display virtuale: 25 o 30 linee di programma, di 40 o 80 caratteri ciascuna, memorizzate ed accessibili, con conseguente visualizzazione sui 16 caratteri del display a LED incorporato nel sistema.
L'operazione si realizza con uno scorrimento verticale ed orizzontale ottenibile con due tasti, tra i 62 presenti sulla tastiera QWERTY. Ma forse il vero punto di forza di questo portatile è costituito dalla gamma di interfacciamenti incorporati, tali da consentire una interessante espansione a partire dal modulo base.
Si noti però che in tal modo si viene progressivamente appannando la "portabilità", al crescere degli oggetti di corredo che decido di usare.
Perciò questo PC come altri di ridottissime dimensioni, presentano nella versione base una portabilità massima, con però prestazioni limitate, potenzialmente incrementabili a ottimi livelli da espansione che ne pregiudicano la portabilità.
D'altra parte abbiamo altri portabili, come l'Epson, l'Osborne 1, che a prima vista (l'Osborne 1, soprattutto) non sembrano poi tanto facilmente trasportabili, ma che comunque mantengono costanti le loro prestazioni, a trasportabilità costante.
Tornando agli interfacciamenti del New Brain, abbiamo due uscite per cassetta audio, una linea seriale per stampante, un'altra seriale bidirezionale, con capacità da 75 a 9600 baud.
Naturalmente è collegabile ad un TV per una visualizzazione completa del display virtuale conservato in memoria.
L'unità centrale è costituita dal microprocessore Z80, unitamente ad una unità slave COP 420 della National Semiconductor. La memoria ROM è 280 Kbyte, più 1 Kbyte nel COP, mentre la RAM è 32 Kbyte. Il sistema operativo è stato realizzato specificatamente per questo PC ma, si parla anche di CP/M, con una scheda di estensione. Il BASIC viene considerato molto potente soprattutto per le sue istruzioni grafiche.
Tornando alla struttura fisica del New Brain, un grave difetto riguarda l'alimentazione, che non è presente nel corpo delle macchine, per cui occorre prevedere un modulo addizionale di alimentazione. Prevista, come opzione, una batteria.

Conclusione

Questa breve panoramica delle novità, voleva essere uno stimolo a prendere in debita considerazione questo ennesimo travestimento dei PC: il portatile. Questa nuova famiglia non è costituita da oggetti che alla portabilità hanno sacrificato prestazioni e comodità d'uso, ma molto spesso sono equilibrate creazioni tecnologiche atte a risolvere momenti operativi particolari, diciamo itineranti, oltre che essere (è indubbio) un affascinante nuovo gadget che vorremmo subito avere. ■

ib INFORMATICA BIELLA

-PROGETTATE

-COSTRUITE

-COMMERCIALIZZATE

-ASSISTITE

DA UN'UNICA AZIENDA LEADER DEL SETTORE

INTERFACCE

Per tutti i tipi di macchine da scrivere elettroniche:
OLIVETTI ET 121
OLIVETTI ET 201/221/231
OLIVETTI ET 225

ADLER SE 1005/1010
ADLER SE 1011/1030
ADLER GABR. 8008

Disponibili in OUT CENTRONICS
oppure IN/OUT RS 232

DISTRIBUTORI REGIONALI

PRODOTTI LEMON II:

PIEMONTE

LIGURIA
VALLE D'AOSTA

UNA SCELTA ECONOMICA E SICURA

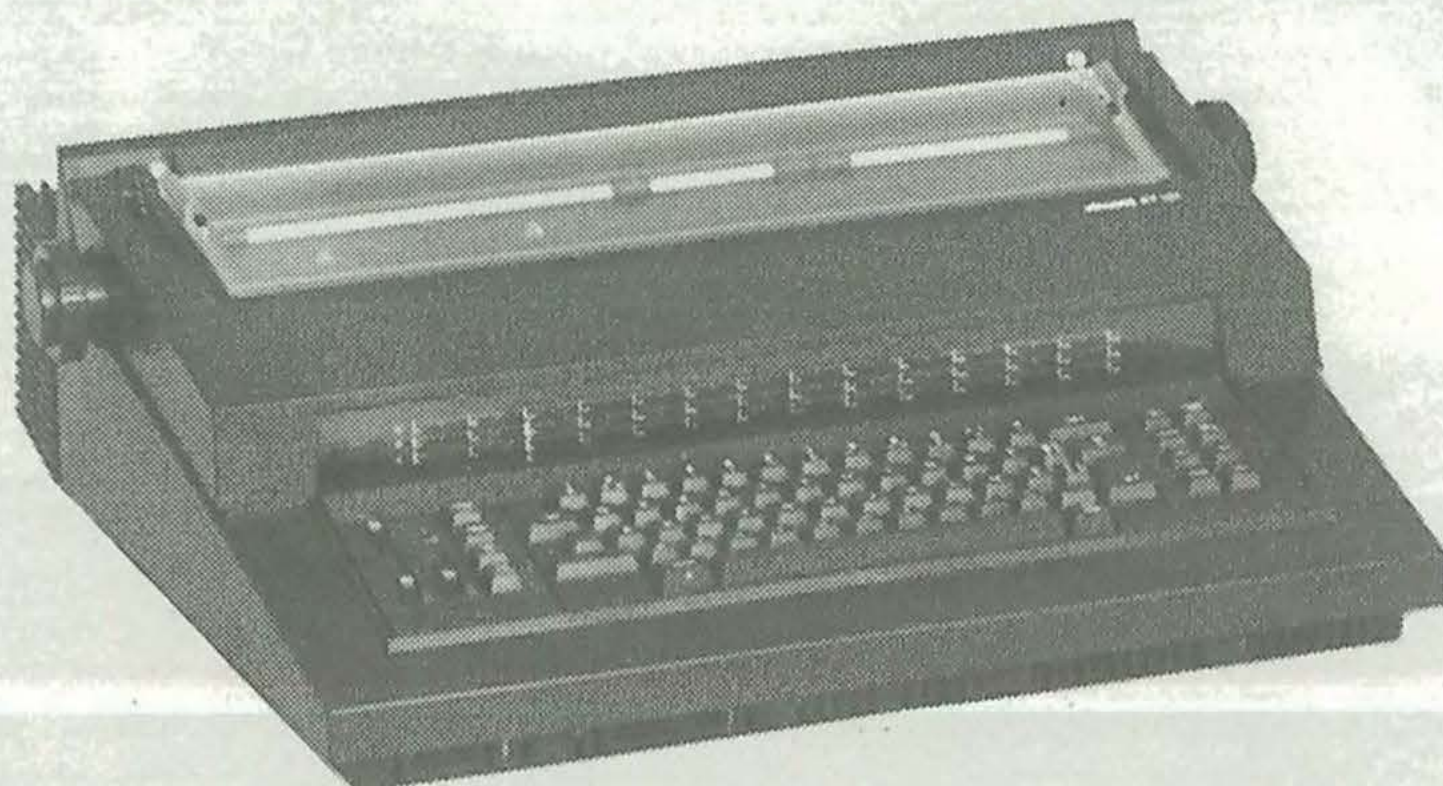
NOVITA'

BIP/121

Il più economico sistema
di gestione testi a partire
da £. 2.600.000 + IVA
(esclusa M.P.S. OLIVETTI ET 121)

Disponibile con programmi:
APPLE* WRITER
Nella versione CPU 6502 LEMON II
WORDSTAR*
Nella versione CPU Z/80 comp. CP/M*

13051 BIELLA - PIAZZA S. PAOLO, 1/B - TEL. 015/29.875 24.181

CORPORATION San Rafael CALIFORNIA
*CP/M é un marchio DIGITAL RESEARCH
PACIFIC GROVE CALIFORNIA
*APPLE é un marchio APPLE COMPUTERS

Mostre e fiere

Romaufficio '83, un grosso successo

di T. Gaudio

Sostenuta dalla presenza di quasi tutti i big dell'informatica, vetrina di piccole e medie aziende del Centro-Sud questa Romaufficio '83 è stata un successo di pubblico e di affari, assai maggiore delle sue quattro edizioni precedenti.
Due ci sembrano i fatti (per la verità scontati) che hanno caratterizzato questa mostra: l'invasione dei registratori di cassa e il successo dei personal computer.
Che sarebbero spuntati Cash Register dappertutto (anche se poco hanno a che fare con l'ufficio) c'era da aspettarselo, poche settimane dall'approvazione di legge che obbliga l'uso del registratore per gli esercizi con giro d'affari superiore ai 200 milioni.

La tastiera di Apple IIe. Vi sono 63 tasti, tutti i segni grafici dei vari alfabeti nazionali, tasti direzionali dedicati al cursore, tasti funzione "mela piena" e "mela vuota", il blocco delle maiuscole e, per gli amanti del tocco lieve, una cura veramente notevole di questo particolare. Da notare che si è tornati al principio delle tastiere dei primi modelli di Apple II, che avevano un supporto con il suo pistoncino a molla per ogni tasto, molto più affidabili, resistenti e piacevoli al tatto che non il modello successivo ora evidentemente messo da parte.

L'interno di Apple IIe, visibile in questa foto, mostra la pulizia fatta sulla scheda madre adottando chip custom che raggruppano molte delle funzioni prima eseguite con un maggiore numero di componenti. Al centro vi è lo slot per l'espansione da 128 Kbyte.

Che il vero protagonista della mostra sarebbe stato di nuovo il personal computer era anche scontato. Folle strabocchevoli, ancora una volta davanti ai "videogame"; "festa" della IBM Italia per il ritorno all'ovile della pecorella smarrita, nuovi modelli, programmi ed accessori. Un personal però diverso già maturo per giocare un ruolo nuovo nella telematica e nell'office automation: non più oggetto fine a se stesso ma elemento integrato nel sistema, terminale intelligente per l'ufficio, capace di sofisticate elaborazioni a livello locale.

Curiosando tra gli stand

Discrete novità nel settore personal, in cui si segnala, come già detto, il Personal IBM presentato dalla IBM Italia Distribuzione Prodotti.
Oltre al suo Personal Computer, anche macchine e sistemi per l'ufficio e sistemi per la gestione aziendale e per il calcolo tecnico-scientifico hanno ca-

Un nuovo terminale della Televideo vantato come rivoluzionario dal punto di vista dell'ergonomia. L'appoggio per le mani davanti alla tastiera è molto ampio e lo schermo è di 14" anzichè 12". Tutta l'elettronica è nella parte destra del mobile, a fianco del cannone elettronico.

```
αβψδεφγηιξκλμνοπ ρστθωςχυζ⊖≠≅∫ϕf0123456789 ∠≮£¼⅓½⅔¾
ABΨΔEΦΓHIΞKΛMNOΠ PΣTΘΩ XYZ⊖Tφ∞●¿¢°%±+×√↑←↓→•

ABCDEFGHIJKLMNOPQRSTUVWXYZ@[\]^_0123456789 !"#$%
abcdefghijklmnopqrstuvwxyz`{¦}~?&'()*+,-./:;<=>

ABCDEFGHIJKLMNOPQRSTUVWXYZ@[\]^_0123456789 !"#$%
abcdefghijklmnopqrstuvwxyz`{¦}~?&'()*+,-./:;<=>

ABCDEFGHIJKLMNOPQRSTUVWXYZ@[\]^_0123456789 !"#$%
abcdefghijklmnopqrstuvwxyz`{¦}~?&'()*+,-./:;<=>
```

Nello stand della Mactronics di Milano era esposta la word scribe WP6000 ultima stampante a matrice della casa americana Anadex. La qualità della stampa è veramente ottima alla velocità di 150 cps, (300 cps con qualità commerciale): in modo grafico indirizza 140 x 140 punti per pollice e il suo costo è di 3.500 dollari. Prossimamente disponibile la versione a colori.

ratterizzato la presenza di un altro big, la Olivetti. Una presenza articolata in due stand (informatica e prodotti per l'ufficio), con parecchie soluzioni per un'ampia fascia di utenza: dagli uffici, all'industria, agli studi professionali e commerciali. Prim'attore anche qui l'M20, arricchito nelle configurazioni e prestazioni, con nuovi collegamenti esterni che lo rendono inseribile in sistemi di office automation e lo aprono ad applicazioni di telematica. Tutto quello che (per ora) è in grado di fare lo si può trovare nella seconda edizione del "Quaderno del Software", distribuito per l'occasione, con oltre 300 programmi applicativi, preparati dalla Olivetti e da un centinaio di software house italiane.

Il sistema hard copier a colori 4691 della Tektronix, basato su tecnologia ink-jet resa assai affidabile e precisa con vari accorgimenti. Si trattava dell'unico modello disponibile in Europa fatto girare un po' in tutte le fiere del settore.

La Honeywell ha invece presentato un nuovo modello multiposto a 16 bit (si chiama Questar/M 9050) che amplia verso l'alto la gamma dei microelaboratori professionali della serie Questar/M introdotta in Italia due anni fa. Il "megamicro", come lo hanno battezzato alla HISI, presenta novità interessanti. Il sistema operativo Prologue 90, notevolmente potenziato, consente ad esempio di operare in tempo reale in multiprogrammazione con possibilità di collegare (direttamente o per via telefonica) fino ad 8 posti di lavoro.

Posto d'onore, nello stand della Hewlett Packard era riservato all'elaboratore HP3000 nella nuova versione serie 40/SI (gestione produzione/materiali e office automation). Nuova filosofia in materia di applicazioni per l'HP125. In un'ottica di sempre maggiore integrazione dei sistemi informativi e di quelli per l'automazione d'ufficio anche all'HP 125 verranno riservati ruoli impegnativi.

Presto, ci è stato detto lo si vedrà inserito nel progetto HP Mail con l'HP3000.

Già annunciato allo Smau, la Saico ha presentato "Grappolo" microcomputer multiutente progettato e prodotto interamente in Italia. Eccolo in sintesi: unità base con Z80A, memoria principale da 64 Kbyte, video 12" (con opzione grafica) e tastiera, stampante seriale o parallela, unità dischi da 2 a 40 Mbyte.

Tra i prodotti presenti allo stand Saico, va segnalato una importante novità, che sarà comunque introdotta ufficialmente sul mercato tra qualche tempo: si tratta di MITO 68000, basato sull'omonima CPU Motorola. Il sistema è dotato di 128 Kbyte di memoria centrale con estensione fino a 16 Mbyte. Supporta sia floppy sia dischi fissi e removibili, con possibilità di indirizzamento diretto fino a 16 Mbyte. Caratteristica fondamentale sarà comunque il sistema operativo Unix.

Altre novità

La Nixdorf ha esposto il suo personal PC 8810 in veste di posto di lavoro intelligente collegato al sistema 8870.

Nuova generazione di elaboratori nello stand Texas. Si tratta della serie Business System 300, una nuova linea di sistemi compatti e multiutenti per applicazioni commerciali.

La Saga, a due anni dall'inizio della sua attività di costruttore, ha presentato il nuovo microcomputer multiterminale F9/8 (unità centrale con due Z80, memoria da 256 Kbyte, floppy da 800 Kbyte e unità a disco fisso winchester da 10 a 20 Mbyte fino a 3 posti di lavoro con video e tastiera).

Due segnalazioni di aziende poco conosciute presentatesi per la prima volta al pubblico di Romaufficio. Dal profondo Sud, la SIDI (Società Italiana di Informatica). Nata negli ultimi anni '70 a Messina, è oggi una delle system house più potenti del mezzogiorno. OEM Honeywell sin dalla nascita, vende ed assiste calcolatori ed altri prodotti informatici. Da Prato, la Centro Matic con "Passi", un nuovo sistema computerizzato per la rilevazione automatica delle presenze, controllo degli accessi e gestione delle mense aziendali.

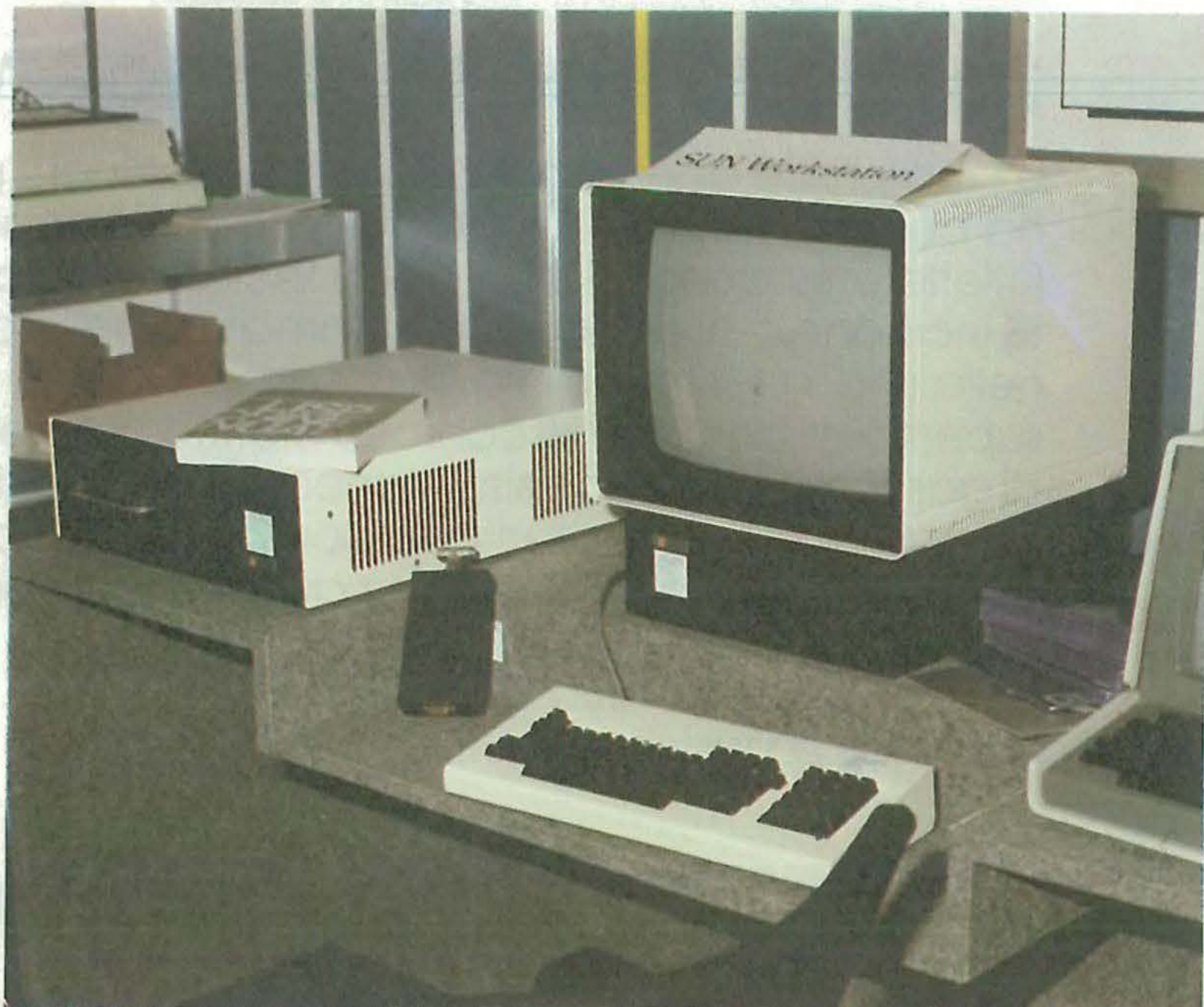
EDP-USA 83, un'edizione in tono un po' minore

di G. Giaccaglini

Lo andiamo ripetendo da un certo tempo che la classica manifestazione del Centro Commerciale Americano, che ha avuto un indubbio merito storico di propagandare l'informatica evoluta (e quella "micro" in particolare) nel nostro paese da diverse edizioni mostra se non proprio la corda, quantomeno qualche acciacco tipico di una gentildonna alquanto matura. Stavolta poi, stretta da vicino tra Romaufficio e, nell'arco di uno stesso mese, la BIAS (a sua volta inglobante Bit '83) la vecchia signora è apparsa un tantino ansimante. Le novità - almeno per gli insaziabili palati di chi si è andato assuefacendo alla droga dell'evoluzione a getto continuo del settore EDP - non erano troppe nè complessivamente sconvol-

genti; detto questo tuttavia saremmo estremamente ingiusti se dicessimo che questa edizione è risultata priva di interesse e chi sa che, magari, il tutto non s'inquadri in una tendenza riflessiva del pur tumultuoso mercato. Oltretutto, secondo quanto non abbiamo mancato di rilevare in precedenti occasioni, EDP-USA per molti operatori ed utenti rimane un'occasione per discutere, trattare, elucubrare ipotesi e realizzare impegni concreti.
Venendo allora, per liquidarle rapidamente, alle notazioni non positive, accenniamo all'assenza, da un lato, della Harden (segno, magari, anche degli assestamenti che, a quanto si dice in giro, preludono ad un forte rilancio della società cremonese?) e, dal lato dei prodotti, di quella LISA - fascinosa e risparmiosa, se ci è consentito prender a prestito aggettivi utilizzati per lanciare ben altro tipo di prodotto suppergiù nello stesso periodo - che la stampa aveva potuto ammirare alcuni giorni prima.
Sul piano delle novità comunque non mancava neanche un oggetto quasi altrettanto mirabolante: la stampante grafica per hard copy di terminale a colori TEK 4691. Nata per riprodurre su carta tutte le sfumature del noto terminale videografico Tektronix 4131, essa adotta la delicata tecnologia del getto d'inchiostro rendendola precisa ed affidabile con vari accorgimenti particolari, tra cui la pulizia dei quattro ugellini per la fuoriuscita degli inchiostri con microgetti d'acqua alla fine d'ogni copia di video. Si paga attorno ai 25 milioni, che già corrispondono ad un discreto rapporto prezzo/prestazioni, ma il prezzo è destinato a divenire più invitante con la futura emanazione di modelli di fascia inferiore. La grafica in genere ha ricevuto un forte risalto e, su questo terreno ci sembra di dover segnalare i nuovi plotter a tamburo 945 e 965 della Calcomp: costano rispettivamente 17 e 26 mila dollari e, muniti di 4 penne, si caratterizzano entrambi per il firmware di controllo incorporato che consente all'operatore d'impostare, su di una estesa tastierina, diverse modalità operative (tra cui il cambio di scala e l'orientamento delle figure) liberando il software del sistema da molte incombenze.
Passando ai sistemi elaborativi, vivo interesse ha suscitato il "superpersonal" a 32 bit della Hewlett Packard HP 9000: atipico rispetto ai supermini "classici" come il VAX della Digital e rivali, perchè si offre come stazione di lavoro superpotente e

La incredibile stampante della serie "E" proveniente dalla linea di prodotti della Control Data, ora acquisita dalla Centronics, con velocità di 1.200 linee al minuto. Qui si vede la macchina aperta, il nastro metallico, appena visibile sulla porzione di puleggia mostrata nella foto, porta tre serie complete di caratteri tra i quali la logica della stampante sceglie la lettera più vicina per utilizzarla in stampa.

La 154 Graphics della Centronics, recente release della casa americana presentata a EDP-USA a Roma. Le principali caratteristiche sono: 120 cps, 132 colonne, matrice di 11 x 8, bidirezionale ottimizzata nel percorso, con capacità di indirizzamento grafico e compatibile a livello di grafica con la 739 Centronics.

La stazione di lavoro Sun della Sun Microsystems, distribuita dalla Delphi di Viareggio. Il microprocessore è un 68000/68010 a 10 MHz, può essere interconnessa in reti tipo Ethernet ed è particolarmente indicata per applicazioni scientifiche o in ambito di progettazione assistita dal calcolatore. Dispone di display ad alta risoluzione e di dispositivo di puntamento a "topolino".

TEMPI DURI PER LA SOFT-FILIBUSTA, CON MICROPRO

Incontriamo Jacques Andreatta della MicroPro International France presso lo stand della torinese Sigesco. Sta contendendo al titolare la versione del WordStar che costoro hanno con fatica italianizzato e che, perciò, non vedono troppo volentieri che possa essere distribuita tramite la casamadre anche a propri concorrenti....
Ad Andreatta chiediamo quali sono le tendenze attuali sul mercato del software-pronto e le linee strategiche che la sua società intende perseguire. Risposta: in Europa il discorso del "logiciel" (si esprime così, da buon francese, anche se di italiche radici) è da un pezzo maturo, Italia compresa. È per questo che la MicroPro, fornitrice di quell'insieme di pacchetti così popolari che, accanto a Word Star, annovera una serie coordinata: DataStar, MailMerge, SuperSort e CalcStar, dall'Agosto '81 ha deciso di impiantare proprie sedi sul vecchio continente. Accanto alla MicroPro France di Parigi che cura l'area sud, in Germania troviamo una MicroPro International che, oltre ad occuparsi del settore nord europeo, si incarica della produzione per l'intero mercato. Peraltro lo sviluppo di nuovi prodotti e versioni viene effettuato in Irlanda (vi lavorano una quarantina di persone).
Una così decisa presenza è finalizzata a far chiarezza oltre che ad accentuare la penetrazione dei MicroProdotti. L'Italia, ci dice il nostro oriundo, ci occupa e preoccupa perchè sta entrando adesso nella fase di maturazione in cui Francia e Germania si trovano da un pezzo, in compenso la situazione è quanto mai ingarbugliata: scarsa chiarezza di rapporti commerciali, assistenza agli utenti sovente inadeguata, pirateria del software che sono i mali non troppo oscuri. Dunque il problema è anche quello di mettere ordine in questo groviglio (Andreatta, nel suo discreto italiano, lo definisce "casino"). Nell'arco di questi mesi egli sta rivedendo i contratti, operando insieme una cernita selettiva dei più qualificati cui si chiede un maggior impegno e professionalità, in cambio di una formazione commerciale e di più tempestivi ausili tecnici che MicroPro fornirà a tutti i distributori. Non verranno rilasciate esclusive a nessuno, anche se la distribuzione verrà organizzata il più possibile al fine di assicurare la massima copertura geografica e insieme le minori interferenze reciproche. Canali di diffusione alternativi a quelli degli OEM o dei fornitori di sistemi sono allo studio: ad esempio in Francia è stato fatto un accordo con un editore di libri tecnici.
È la guerra per debellare la Filibusta del software? Premesso, dice Andreatta, che valide difese passive non ne esistono dato che (consoliamoci) non solo in Italia ma in tutta Europa si hanno carenze legislative assolute in materia, la lotta verrà effettuata nel quadro della riorganizzazione già detta, approffittando anche dell'assai facile e rapida circolazione di notizie entro l'ancor piccolo villaggio microinformatico. Si farà capire che comportamenti sleali non pagano più in convenienza per chi desidera mantenere rapporti, lavorando in buona armonia, pena la perdita della concessione nei casi più gravi o recidivi. Verranno comunque prese per misure indirette, come l'adozione di un formato non standard per i fogli dei manuali (per rendere quantomeno più onerosa le fotocopiature) e l'incisione sulle floppy-copie di un numero di serie che deve essere segnalato alla casamadre. In tal modo questa, se da un end user in buona fede, ricevesse la richiesta di una nuova release può risalire facilmente al misfatto e a chi lo commise (è la trafila del caso che, negli USA, ha dato origine ad una pesante sentenza di condanna, per la prima volta in questo campo).
Quella delle versioni in più lingue (francese, italiano, spagnolo, portoghese, tedesco per ora) è uno degli impegni più pressanti della MicroPro, perlomeno con i pacchetti più diffusi: WordStar e MailMerge verranno tradotti per l'85 - 90% sia dei manuali sia dei messaggi ad eccezione di quelli di sistema). Al momento in cui quest'intervista è pubblicata tali traduzioni dovrebbero già essere terminate. Su nuovi prodotti e/o release d'ora in poi MicroPro assicura l'emissione delle versioni plurilingue nell'arco massimo di 8 settimane dalla nascita.
È pure in arrivo un nuovo, potente package. Si chiama InfoStar ed è destinato a sostituire l'ormai vetusto DataStar. Si pone come emulo dell'affermato dBase II - in procinto di diventare un bestseller anche da noi - vantando, almeno a detta del nostro, superiore qualità in termini di velocità e, soprattutto, di facilità d'uso. Interessante particolare, chi possiede DataStar può evolverlo in Info.... lasciando immutata la Star: non certo, ovviamente, ricorrendo a questo sciocco espediente da Settimana Enigmistica, ma impiegando una opportuna utility di metamorfosi.

munita di Unix. Questo sistema operativo, a giudizio di diversi osservatori, brillava stavolta per la sua forte presenza su molte macchine a 16 bit tra cui un'altra novità interessante presente al padiglione della Zelco. Stiamo parlando del Fortune 16:32 che, come le cifre che accompagnano il suo nome denotano, adotta come CPU l'MC 68000 il cui bus indirizzi è di 32 bit. All'Unix esso associa un hardware assai curato, una linea filante ed assai ergonomica e software piuttosto potente. Personalmente abbiamo potuto apprezzare un package di word processing che fa diverse cosette in più del consueto, come ad esempio l'uso di fino a 4 finestre-video indipendenti per redigere testi distinti e/o variamente correlabili. Questa supremazia di Unix faceva l'evidente felicità di un pioniere in materia come Giulio Bertellini che a suo tempo importò quel fortunato sistema Onyx e che, in questa occasione, ci ha pure espresso la sua convinzione che anche questo "nobilsistema" operativo è destinato ad una larga popolarità, anche al di fuori degli ambienti paludati ed aristocratici presso i quali si è finora per lo più affermato.
Altri spunti che, in chiusura, ci sentiamo di accennare (rimandando alle foto per una piccola serie di flash su altri prodotti) si riferiscono all'importanza crescente del software standard, testimoniata dall'ingresso con mire, a quanto sembra, aggressive, della MicroPro fornitrice dei noti package come WordStar, CalcStar, MailMerge ecc. nonchè all'emergenza ormai netta dei sistemi a rete locale. Su questo tema, come su quello più generale delle telecomunicazioni EDP-USA 83 si è rivelata piuttosto ricca, come sta avvenendo ormai da diverse edizioni.

BIAS '83

di G.M. Menegardo

La diciottesima edizione del BIAS tenutasi quest'anno a Milano ha fatto registrare, contrariamente alle previsioni della vigilia, un ottimo successo di pubblico.
La rassegna, dedicata quest'anno alla microelettronica, riservava, come ormai consuetudine, un cantuccio ai personal computer. Sta diventando una abitudine tra i frequentatori di questo tipo di fiere specializzate lo scambiarsi le impressioni sul futuro o sui propri affari o su come va il mercato: in genere sono le chiacchiere che si fanno nei momenti di "stanca", quando i visitatori ancora non affluiscono in massa, normalmente il primo giorno al mattino, oppure la sera dopo la chiusura dei cancelli. Cosa ci si dice in queste occasioni? Tra i professional più di uno "mugugna" dicendo che non si può più lavorare, intendendo che le ore passate a curare i ragazzetti che armeggiano sulle macchine esposte non sono adeguatamente ricompensate con ritorni in vendite o contatti d'affari come ci si aspetterebbe, specie dopo avere pagato il salato affitto dello spazio espositivo. Si avanzava la proposta di far pagare l'ingresso in modo da selezionare maggiormente il pubblico. Anche il catalogo è stato criticato per via del fatto che espositori e ditte esposte figuravano tutti nei medesimi capitoli, aggiungendo molta carta ma nessuna informazione in più. Qualche nota società non era presente, rendendo evidente la scelta, che altri seguiranno, di selezionare accuratamente le fiere alle quali partecipare.
Ma veniamo alle novità presenti. Poichè tutte le aziende con un riferimento ai personal erano confinate in un unico luogo, è stato agevole fare "il giro". Subito dentro il padiglione 18 c'era la Lorenzon con il suo Shine, un home computer di release non recentissima, la cui presentazione avvenne infatti al BIAS del 1981, che però era equipaggiato con una scheda per la sintesi vocale che ne costituiva un po' la novità. Chi ha seguito il TG1 delle 13.30 di giovedì 24 Febbraio, ha potuto sentirne la buona qualità di dizione. Il programma permetteva all'operatore di digitare una stringa di caratteri dalla tastiera che dopo aver premuto il tasto *RETURN* venivano ripetuti dalla voce baritonale sintetizzata dal chip contenuto sulla scheda vocale collegata alla porta seriale dello Shine. Si tratta per il momento di una attrazione da fiera, appunto, i cui usi pratici saranno più chiari in seguito. Poco più avanti, nello stand occupato dalla Olivetti, si sentiva risuonare un'altra voce metallica: era quella dell'M20 che autopresentava le caratteristiche del suo nuovo schermo a colori con la dimostrazione di alcuni grafici molto variopinti.

Nello stesso stand l'M40 lavorava in completa emulazione del P6060, un minicomputer della casa di Ivrea da alcuni anni sul mercato, di cui poteva quindi utilizzare il software disponibile. Più oltre c'era lo stand della Harden, distributrice per l'Italia del Sirius, che mostrava su un mobile la stampante Microline 84 della OKI con un nuovo inseritore di fogli singoli da pacco, e avvertiva di poter disporre entro breve di due altri prodotti, uno di classe inferiore e uno superiore al Sirius, che rimangono per ora segreti. Lo stand Celdis esponeva un Rainbow, il personal computer della Digital Equipment, assieme ai due sistemi professionali 250 e 350 e alle stampanti con accoppiatore acustico incorporato LA40 Correspondent, sempre della Digital. Alla SEGI il ruolo di primattore spettava all'HX 20 della Epson, piccolo compatto con stampantina incorporata che già è stato presentato in altre occasioni fieristiche, ma che qui era interfacciato a un termometro analogico da 100 MV/Centigrado e a un convertitore A/D a 8 bit di produzione della Symic. La ICL presentava il suo noto personal computer e la stazione di lavoro intelligente PERQ, mentre un poco discosto la Digitek esponeva il MPFII, acronimo che sta per Micro Professor, con una tastiera staccata e più grande di quella sull'unità centrale, il che rende perciò il lavorarci più agevole. Questo microcomputer è appena arrivato da Taiwan, con CPU 6502, e già due diverse società pretendevano entrambe di esserne le esclusiviste per l'Italia. Comunque la Digitek è ora quella ufficiale approvata dalla Multitech, la casa costruttrice, come risulta da un telex di pochi giorni prima del BIAS che veniva mostrato ai giornalisti di passaggio. Presente in più stand il Personal Computer IBM, fino all'ultima fiera disponibile solo presso la Hard & Soft di Rimini, ma ora a disposizione sia della Eledra che della DPI. Nello stand Eledra il solito Lemon II e alcuni altri compatibili: il Columbia della Data Product verso l'IBM e il "Franklin Ace 1000" verso Apple II (di cui rappresenta una versione espansa). Dello stesso distributore anche il Kontron

Sempre allo stand Mactronics era visibile, (agli intimi) questo piccolo terminale portatile, denominato Husky e prodotto dalla DVW Microelectronics.

Psi 80 D. Alla Telco il Voyager 100, portatile CP/M Osborne-like, distribuito anche dalla Eltron di Brescia che però non era presente. Alla Ecta di Milano oltre a un plotter Watanabe era visibile un microcomputer denominato Viterm 520; più avanti, allo stand della Rebit Computer, era esposto il Sinclair ZX-Spectrum che la direzione commerciale della società, nella persona del Sig. Fontana, conferma che sarà disponibile in quantità già al momento di leggere queste note. Questo home computer può essere reperito anche presso i Bit Shop Primavera. Anche la Kyber esponeva dei prodotti di cui la novità era rappresentata dalla versioni del Minus, il personal computer della casa di Pistoia, con memoria di massa configurata con dischi winchester da 5, 7, 10 e 15 Mbyte. Interessante il back-up di hard disk da 15 Mbyte con microwinchester removibile da 5 Mbyte. Anche presso la Melchioni Computertime c'era qualcosa di nuovo, ma mentre per quanto riguarda le macchine da scrivere elettroniche e un sistema gestionale denominato Hayac 2900 tutto era alla luce del sole (se così possono assimilarsi le lampade al neon), per quanto riguarda un nuovo personal computer della Sharp il discorso è decisamente privato. Sperando che non arrivi un fulmine in redazione ne pubblichiamo l'indiscrezione. Si tratta di una macchina CP/M denominata MZ-3541 con due Z 80A con processore per stampante dedicata con spooling di stampa. La RAM è di 128 Kbyte, necessari quando si carica il BASIC esteso che occupa ben 61 Kbyte; lo schermo ad alta risoluzione ha una struttura di 640 x 400 punti ed è predisposto per il colore.

Un ultimo stand presente era quello della Logical Machines, una azienda che vende elaboratori di medie dimensione per gestionale programmabili in lingua italiana.

Questo è quanto c'era al padiglione 18; nel padiglione 15 invece dopo un po' di giri poteva capitare di imbattersi in un'interessante applicazione dei personal nel settore degli allarmi, presentata da una piccola azienda di Monza, la Hardware.

Un VIC 64 controllava tramite la user port 64 periferiche, collegate serialmente da tre fili, di cui rilevava lo stato logico.

Ognuna delle periferiche era configurata con dei dip-switch e aveva inoltre la possibilità di rilevare 16 tipi diversi di segnali. La scatola di controllo con l'elettronica eseguiva un polling delle diverse periferiche e, trovando una situazione anomala, inviava il segnale di ritorno al microcomputer che lo utilizzava sia facendo partire delle routine di allarme, sia l'uscita a video per avvertire un eventuale operatore. Oltre questa applicazione piuttosto notevole (i tre fili che collegano i periferici possono essere lunghi fino a 1.000 metri), un prodotto degno di nota era un cabinet per insonorizzazione delle stampanti a impatto, prodotto dalla Viking e esposto dalla Ragom di Milano.

Nel padiglione 14 presso lo stand della Siprel era annunciato un nuovo Apple compatibile seppure di costo molto contenuto, denominato Personal Kid, proposto sia in versione open frame che in cabinet.

Alla Telcom era esposto un nuovo terminale video carrozzato da Bertone, più noto quale designer di carrozzerie automobilistiche.

Questo è il resoconto del BIAS dal punto di vista più vicino all'utente; parallelamente e più per gli addetti ai lavori, si sono svolti tre diversi incontri di studio organizzati dal Fast, dalla rivista **Elettronica Oggi** e da Reseau. A parte il successo di questi ultimi, c'è da rilevare, a proposito dei mugugni degli addetti ai lavori sopracitati, che l'assalto dei giovani avventori così poco considerati nelle discussioni serali tra professional, sono ancora la pietra di paragone sulla quale si misura il successo delle manifestazioni.

Tanta gente e tanta confusione potranno dare fastidio, ma forse in nessun altro modo che questo ci si fa un'idea di come sia il mercato; il resto sono lamenti probabilmente destinati a sollevare solo sospetto.

Mostre e fiere

Stimolante convegno dell'AIPI

di G. Giaccaglini

Che il convegno AIPI del 23 Febbraio a Milano vertesse sul tema de "L'informatica in tempi di povertà" era cosa che poteva sorprendere chi conosce l'Associazione Italiana Professionale di Informatica come un sodalizio un poco èlitario, (ma forse il fatto che esso sia statutariamente alieno da ogni fine corporativo spiega l'audacia della scelta. Spregiudicato era anzitutto il fatto stesso che si usasse il crudo e francescano termine in luogo di quello, edulcorato, di "austerità" che adottano i politici, ma poi, com'era non solo dietrologicamente scontato, la maggioranza degli interventi ha finito per focalizzarsi sulla crisi di ruolo degli uomini dell'EDP (tradizionale), sulla loro difficoltà anche umana di riconvertire la propria mentalità e professionalità, sull'onda montante delle rinnovate tecnologie. Strettamente connesso, in modo talora indiretto ma spesso più che trasparente, è risultato anche il problema del potere aziendale: lo sviluppo di nuovi tool hardware e software, quello che, nel sommario stesso del convegno, veniva esplicitamente chiamato il fenomeno delle "informatiche parallele", il coinvolgimento sempre più stretto dell'utilizzatore finale si intrecciano strettamente con un processo di redistribuzione dei ruoli nell'ambito dell'azienda. Come è forse fatale, stando alla nota (e citatissima, nel corso degli interventi) Curva di Nolan, stiamo vivendo un periodo disordinato e selvaggio, caratterizzato dalla proliferazione sovente incontrollata dei mezzi più vari. In queste condizioni auspicare, come i più hanno fatto, un'accelerazione della loro integrazione è apparso a taluni piuttosto un'enunciazione di astratti principi. Una nota curiosa: in almeno tre-quattro relazioni venivano utilizzate slide pressochè identiche, tra cui, popolare quant'altre mai, quella delle trepalle: Telecom, Data Product e Office Product che a regime s'intersecano nel mitico trifoglio avente al centro l'Ufficio del Futuro. Al di là dell'imbarazzo un po' umoristico della situazione non è essa un sintomo dell'acquiescenza culturale acritica di troppi nostri manager e consulenti a schemi d'oltreoceano importati?

La fastidiosa impertinenza dei personal

Nella sua introduzione il Dr. Mazza, forte della sua esperienza maturata negli USA, ha fatto un quadro ampio e documentato della situazione. Molto brillante (ha parlato del nuovo "villaggio elettronico" caratterizzato dal telelavoro a domicilio, ha citato esperti vari, sia i pessimisti che vedono nell'informatica e nel suo dilagare pericoli di alienazione, sia chi, invece, la concepisce come apportatrice d'una nuova Arcadia, d'una qualità della vita migliore) ha molto insistito sulla telematica e, soprattutto, sul fenomeno personal computer. Forse è stato anche questo forte accento sull'"uomo dell'anno" di Time che ha contribuito a creare, specie nel dibattito dei numerosi partecipanti, un clima di acceso protagonismo di questo oggetto, ben al di là delle intenzioni degli organizzatori e dei relatori ufficiali. L'impressione che ne hanno ricevuto i cronisti è che, tra le diverse informatiche parallele sorte come funghi a fianco di quella dei sancta sanctorum tradizional-mainframistici: reti a valore aggiunto; package acquistati dall'esterno; nuovi prodotti hard/soft per l'office automation (e/o, nelle fabbriche, per il CAD/CAM) quella proliferante dei personal è senz'altro la più fastidiosa per la più parte degli EDP-manager.
Nel corso del dibattito, accanto al rendiconto di esperienze interessanti, volte tutte all'accettazione gradualistica di nuovi strumenti ed alla modifica intelligente dei ruoli degli EDP manager, si sono manifestate posizioni piuttosto contrastanti, specie sul tema dell'informatica che più sfugge ai controlli: quella dei personal acquistati

autonomamente da dirigenti e tecnici nei vari reparti (stanchi dei disservizi e della lentocrazia del sacro Centro, è stato ammesso da qualcuno...) per l'appunto.
Così il Dr. Floridia, responsabile dei sistemi informativi della RAI, dalla premessa relativa all'esigenza di accelerare la disponibilità dei dati aziendali per tutti gli utenti ha fatto toccare con mano la possibilità di raggiungere gradualmente l'obiettivo della coesistenza pacifica, anzi cooperante di mezzi vecchi e nuovi, insistendo sulla necessità che il professionista d'informatica moderno si orienti alla riconversione di se stesso verso il ruolo di information manager, fornitore di know how a tutta l'azienda più che di sacro tesoriere dei dati (e senza complessi verso le nuove tecnologie di cui è l'unico a possedere il background necessario). Interessante si è rivelato il caso FIAT IVECO, come esperienza di riuscita integrazione di workstation CAD/CAM e sistemi di office automation, scelti questi ultimi in collaborazione con il management: anzi la gestione è suddivisa, nel senso che ad esempio quanto attiene all'automazione d'ufficio ed al personal computing è prevalentemente affidata agli utenti, in un'ottica di decentramento.
In generale l'idea del "personal computing", categoria che tecnicamente si realizza per lo più tramite terminali sui quali l'utente finale dispone di servizi centralmente predisposti, è parsa assai popolare tra la maggioranza dei convenuti. Taluni di costoro hanno persino tuonato, con accenti quasi passionali, contro la degenerazione della spesa dei personal, invocando addirittura misure restrittive e "pianificazioni" rigorose. A chi scrive tali argomentazioni non sono parse del tutto convincenti: a parte il fatto che si finisce col pretendere chiusure di stalla quando già molti buoi sono scappati tranquillamente, è proprio dogmaticamente vero, sempre ed ovunque, che personal a sè stanti siano uno spreco ed un'anarchica rovina? Non si andrebbe così contro quell'autonomia decisionale della periferica che in molte aziende fa parte da tempo dello stile del management?
Pregnante sotto questo profilo ci è parso l'intervento di Vincenzo Monaci della Sistems & Management quando ha invitato i presenti a fare meno COBOL e ad aprirsi sempre di più verso le nuove realtà dell'informatica distribuita, a rivolgersi a sistemi operativi moderni come Unix, ed alle piccole macchine, a dialogare finalmente con il resto dell'ambiente aziendale.
In tutto questo fervido dibattito è rimasto in ombra il tema della povertà (se non come spunto: i minori investimenti di questo periodo recessivo favorirebbe l'acquisto dei microcomputer a danno dei mainframe, ha detto qualcuno). E infatti il punto è un altro: anche sul fronte della tecnologia la crisi è nei valori. ■

Burroughs e NCR presentano due nuovi personal computer

a cura della **Redazione**

Il personal NCR

La società americana NCR ha presentato recentemente il suo ultimo prodotto che si colloca nell'area del personal computing.
Il nuovo personal, denominato Decision Mate V, sino dalla sua nascita denota di avere alle sue spalle una accorta strategia poichè assomma in sè tutte le caratteristiche più desiderabili su un personal computer secondo gli standard di oggi.
Due scelte operate dalla pianificazione di prodotto della NCR sono le più logiche: il doppio microprocessore, disponibile come opzione, che rende portabile su questo personal tutto il software sviluppato per lo Z80 e per l'8088. Inoltre la possibilità di interconnessione di un certo numero di questi personal computer tra loro con una rete locale del tipo Omninet inserisce l'unità nel concetto di informatica distribuita per le medie e grandi organizzazioni.
La possibilità del collegamento in reti di personal computer consiste nell'inserire su un doppino non schermato sino a 63 utenti, di cui 16 contemporaneamente possono essere attivi, e il cui accesso alle risorse comuni è regolato da un file server denominato *MODUS*.
Questo apparecchio si accerta che non vi siano accessi concomitanti ai medesimi file, e costituisce inoltre una porta di comunicazione, con i protocolli più svariati, verso altri elaboratori centrali.
Oltre a questa impostazione di base, sono rimarchevoli nel DM V la possibilità di espansione della memoria centrale, che può giungere a 512 Kbyte con l'aggiunta di moduli di memoria, e l'alta risoluzione del video monocromatico da 12" che visualizza in modo grafico 640 x 400 punti.
È possibile anche realizzare l'interfacciamento verso periferiche diverse dalla stampante (per cui è disponibile una porta parallela) con dei moduli di in-

Il nuovo Burroughs B21 con memoria di massa da 256 a 640 Kbyte.

Mettete il Vostro microcomputer in condizioni di stampare

La Facit 4510, stampante seriale a matrice, 80 colonne, basso costo, è il "purosangue" delle microstampanti. Realizzata per avere una periferica di qualità ed alta professionalità, la Facit 4510 ha come standard tutte quelle caratteristiche che gli altri forniscono come opzioni. La tecnologia a microprocessore e la memoria tampone di 2K caratteri permettono di trasferire dati alla massima velocità consentita dal computer. La versatilità di collegamento è garantita dalle interfacce RS232-C e parallela, entrambe presenti come standard.

Grafica a blocchi ed a punti, assicura flessibilità d'impiego e ottimizzazione del sistema.

Nello standard sono comprese diverse fonti di caratteri, alta risoluzione e 8 repertori di caratteri nazionali.

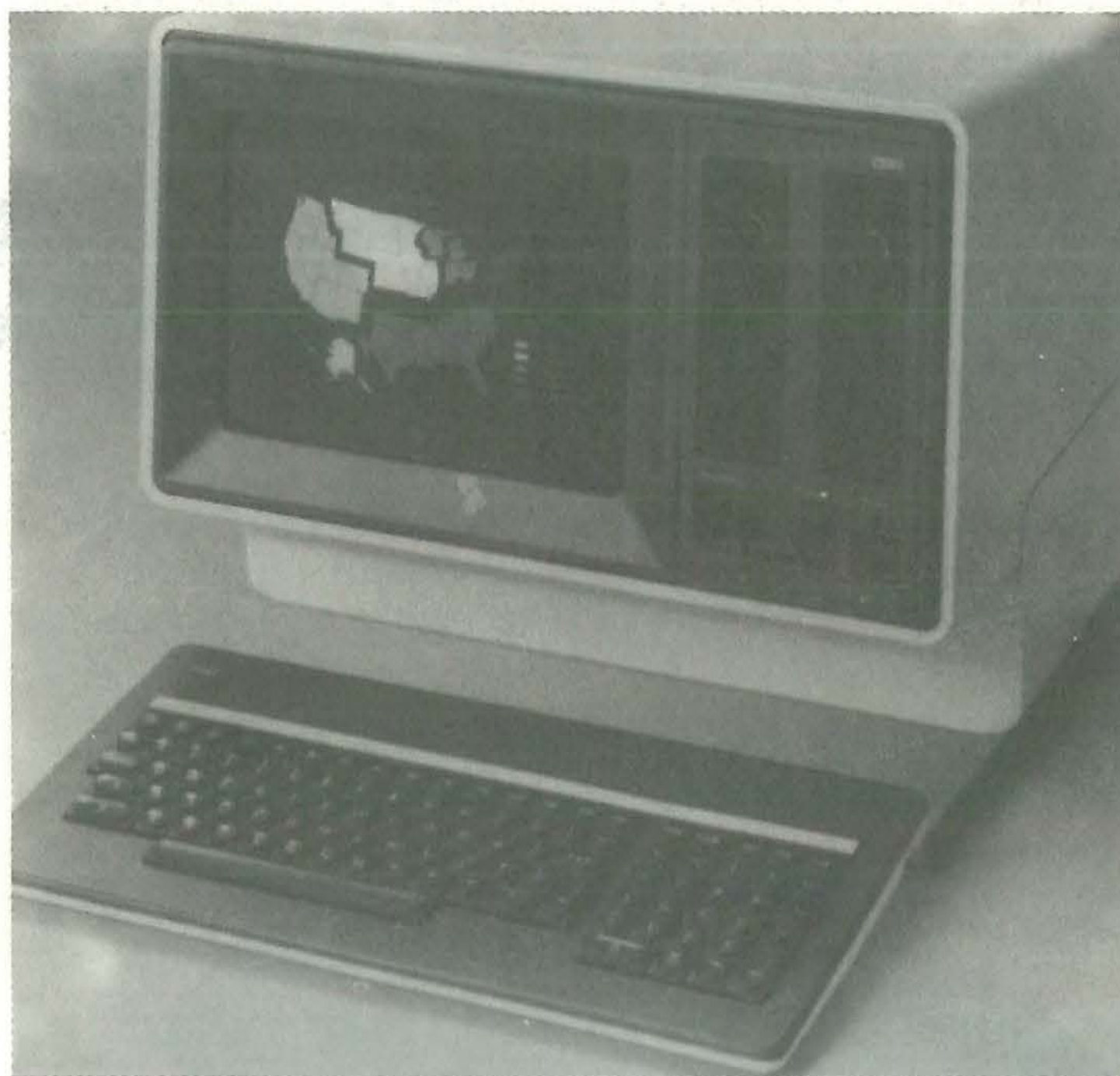
Trascinamento a trattori, carta in rullo tipo telex, foglio singolo, sono i tre tipi di carta utilizzabile che fanno delle Facit 4510 un vero "purosangue".

FACIT DATA PRODUCTS

Via Toffetti, 2 - 20139 MILANO (ITALIA) - Tel. (02) 5694245-6-7-8-9

207

Burroughs e NCR presentano due nuovi personal computer

terfacciamento da inserire nei connettori dedicati sul retro dell'unità cui fanno capo le interfacce seriali RS 232C.
La tastiera a profilo basso separata dal resto dell'unità dispone, oltre al normale set di caratteri e al tastierino numerico separato, anche di 20 tasti funzione definibili dall'utente.
Le memorie di massa comprendono due dischi da 5" incorporati nel mobile del video la cui capacità formattata è di 320 Kbyte, la stessa dei nuovi dischetti del Personal Computer IBM.

Il personal computer Decision Mate V della NCR. Caratteristica saliente è la possibilità di collegamenti in rete di personal.

Nella foto il nuovo Burroughs B21, unità centrale da 16 bit, video, tastiera e opzionalmente la memoria di massa nella stessa struttura da tavolo.

Un particolare della tastiera studiata per fornire all'operatore il massimo confort di utilizzo.

È tuttavia possibile accrescere il numero di byte in linea montando nel mobile un microwinchester. Questo aggiunge 10 Mbyte formattati alla capacità totale. I sistemi operativi che possono essere utilizzati dal Decision Mate V sono, oltre al CP/M 2.2, il CP/M 86 e l'MS-DOS.
La NCR dispone attualmente di un programma per la pianificazione finanziaria, di uno per l'elaborazione dei testi e di un altro per lavorare con grafici vari.
Alla presentazione avvenuta all'Hotel Gallia di Milano, il Dr. Piero Toso, DG della NCR Italia, ha promesso che gli annunci non sono finiti e quanto prima si riparlerà di nuove release della sua società, decisa a rimanere tra i leader nel settore dell'EDP.

Burroughs B21

SLa Burroughs lancia anche sul nostro mercato una ampia gamma di microcomputer dalle prestazioni professionali, suddivise in sotto-serie che vanno dai "superpersonal" è sè stanti (stand alone) a veri e propri sistemi SBS (Small Business Systems). Dalla linea estremamente elegante e, insieme, ergonomicamente attenta ai fattori dell'ingegneria "umana" (videoschermi assai stabili e riposanti, per dirne una) hanno memorie di massa che vanno dai 256 ai 512 Kbyte (nella serie B21) ai 640 Kbyte della serie B22 che parte con 384 Kbyte, mentre la CPU è una Intel 8086 a 16 bit. Molto ampia e flessibile è la configurabilità delle memorie di massa: i drive sono contenuti nell'alloggiamento dell'unità centrale (che si trova a lato dell'unità video, facendole da pandent elegante quanto inusuale, v. foto) e possono essere floppy disk o winchester da 5 pollici e 1/4 con capienze, dopo formattazione, che vanno dai 630 Kbyte dei floppy ai 5, 8.4 o 13.4 Mbyte degli hard disk.
Sulla serie B22 sono disponibili in aggiunta (o in alternativa, la cosa non ci è ancora ben chiara) memorie di massa a parte, disposte verticalmente sotto la scrivania che supporta il sistema, costituite da winchester ancor più capiente con dischetto di back-up da 8 pollici.
Ma dove questi microelaboratori presentano un punto di forza è nella abbondanza di interfacce incorporate o opzionali, arricchite dal software di comunicazione per ogni tipo di protocollo sia per elaboratori Burroughs che IBM. Ci segnalano in particolare le interfacce seriali RS232C ed RS422 le cui velocità rispettive possono raggiungere, nei due casi, i 19,2 ed i 615 Kbaud (serie B22, nella B21 le cifre sono più basse) e l'interfaccia parallela a 9.600 car/sec. (massimo). I protocolli di comunicazione includono l'Asynchronous Terminal Emulator, l'IBM 3270 e 2780/3780, oltre all'X.25 (verso reti pubbliche per dati, come la già varata Itapac del nostro paese) e, naturalmente, al Poll/Select della Burroughs.
Con simili numeri i B20 - che tra altre "feature" che qui citiamo a caso possiedono interconnessioni in DMA ed un proprio sistema operativo multitasking assai potente, benchè non standard - sono apertissimi al dialogo.
Sono a tal fine previste configurazioni a "grappolo" (con dei B20 "smemorati", ossia deprivati di RAM, che attorniano, come terminali, master B22 e loro risorse pregiate) o altre topologie più complesse in cui, salvaguardando l'autonomia elaborativa locale si può accedere a dati siti altrove, anche su minicomputer o mainframe. Questo concetto di informatica distribuita (cui, stando almeno alle prime informazioni, non dovrebbero mancare il sostegno di una "rete locale ad alta velocità" anche se al momento di andare in macchina non è ancor chiaro di cosa si tratta) già si arricchisce di un package per l'office automation (un word processor vantato come assai potente) che si affianca ad una ricca messe di linguaggi di programmazione (COBOL, BASIC, FORTRAN e Pascal), di "tool" di sviluppo programmi, di debugging nonchè un potente sistema di gestione basi di dati ISAM multichiave che permette l'accesso indexed-sequential sulla base di fino a 100 chiavi e gestione "shared" di file con capacità di blocco al record. Insomma le potenzialità di un classico mini gestionale ci sono tutte o quasi, ma a prezzi più contenuti.

■

IL PERSONAL COMPUTER IBM IL TUO PICCOLO GRANDE AMICO.

Crescere da soli non è facile: quante volte hai desiderato l'aiuto di un amico?
Ora l'hai trovato: è il Personal Computer IBM.
Vedrai, imparerai da solo a dialogare con lui in poche ore. Darà sempre una risposta immediata ad ogni tua richiesta e, se vuoi, ti collegherà anche ad altri elaboratori.
Potrà aiutarti in tutte le tue attività.
Il Personal Computer IBM, infatti, controlla tutti quei lavori che, in un momento di crescita, rischierebbero di occuparti troppo tempo. Può fare di tutto: riceve dati, analizza, calcola, registra, stampa e, grazie alla sua potente memoria e ai minidischi, ti consente di gestire un'infinità di informazioni. Ti sarà più facile formulare preventivi e offerte, senza perdere d'occhio il tuo margine di profitto.
Ogni cosa sarà più semplice, con un amico così.
Vuoi conoscerlo meglio? Rivolgiti ai concessionari IBM.
Avrai tutte le informazioni necessarie.

Unità di elaborazione: contiene un microprocessore a 16 bit - ROM di 40 Kbyte - RAM da 64 a 640 Kbyte - una o due unità minidischi oppure una unità minidischi e un disco fisso da 10 Mbyte - architettura aperta - 5 o 8 attacchi per unità periferiche. **Unità di espansione:** contiene uno o due dischi fissi da 10 Mbyte ciascuno (capacità massima del sistema: 21 Mbyte in linea) - 8 attacchi supplementari per unità periferiche. **Tastiera:** ultrapiatta - contiene un microprocessore. **Unità video:** visualizza 25 righe di 80 caratteri con una matrice di 7x9 punti. **Stampante:** grafica bidirezionale - velocità massima 80 cps - 12 tipi di caratteri - densità di stampa da 40 a 132 caratteri per riga. **Sistema operativo DOS:** contiene avanzate funzioni logiche per un'alta flessibilità operativa e una grande facilità d'uso.

Il Personal Computer IBM viene venduto attraverso una rete di Concessionari IBM, costituita a questo scopo da una nuova società, la IBM Italia Distribuzione Prodotti S.r.l. Per acquisti superiori alle 20 unità puoi rivolgerti anche alle filiali IBM. Per informazioni sugli indirizzi dei punti di vendita telefona al 02/21752360 oppure 06/54864962.

a cura della **Redazione**

Bitest: AVT COMP 2

L'ideale è lo scopo a cui si tende, ragione della propria esistenza. Riconosciutolo, ogni azione non può che essere compiuta, se non altro come sforzo continuo seppure a volte fallace, per la realizzazione dello stesso. Un dubbio: se ciò vale per ogni uomo, salvo che per i danteschi ignavi, varrà anche per i personal computer? Domanda buffa e, ne siamo certi, senza fondamento. Senonché, ormai, chi è portatore d'opinione, e la rivista Time certamente rappresenta un importante elemento dei cosiddeti mass-media, ha avallato un modo di concepire queste "stupide" macchinette che si avvicina molto alla vera idolatria. Un uomo modellato in cartapesta, o forse in gesso, volge lo sguardo a un tavolo rosso dalle fattezze antiche, sopra il quale poggia un originale personal computer dotato di evidente grafica a colori e di caratteristiche stilistiche ignote al mercato.

Una copertina famosa, che ha generato non poche polemiche considerazioni. L'interesse di un **Bitest** non poggia comunque su considerazioni e confronti filosofeggianti. Nel caso in esame, d'altronde, parrebbe ancora valido il dubbio posto in apertura. Supponete che la macchina personal computer possa esprimere un desiderio: imitare un suo simile. Quale sceglierebbe? Il più versatile, il più veloce, il più ammirato, il "fusto" con parecchi K di memoria o, vanità delle vanità, il portatore di nome femminile che tiene al guinzaglio un piccolo "mouse"? Fantasticando ancora la risposta è difficile. Come verrebbero valutate queste caratteristiche? Eppure nella pratica un personal, che potremmo definire come quello più imitato del mondo, esiste ed è a lui che molti tendono. Sono le sue possibilità ma soprattutto l'abbondanza di software che lo può alimentare, ad essere invidiate dai più. Complessivamente, potremmo dire, le sue specifiche.

La AVT definisce il suo nuovo personal "pienamente compatibile con le specifiche Apple". Capita l'antifona?

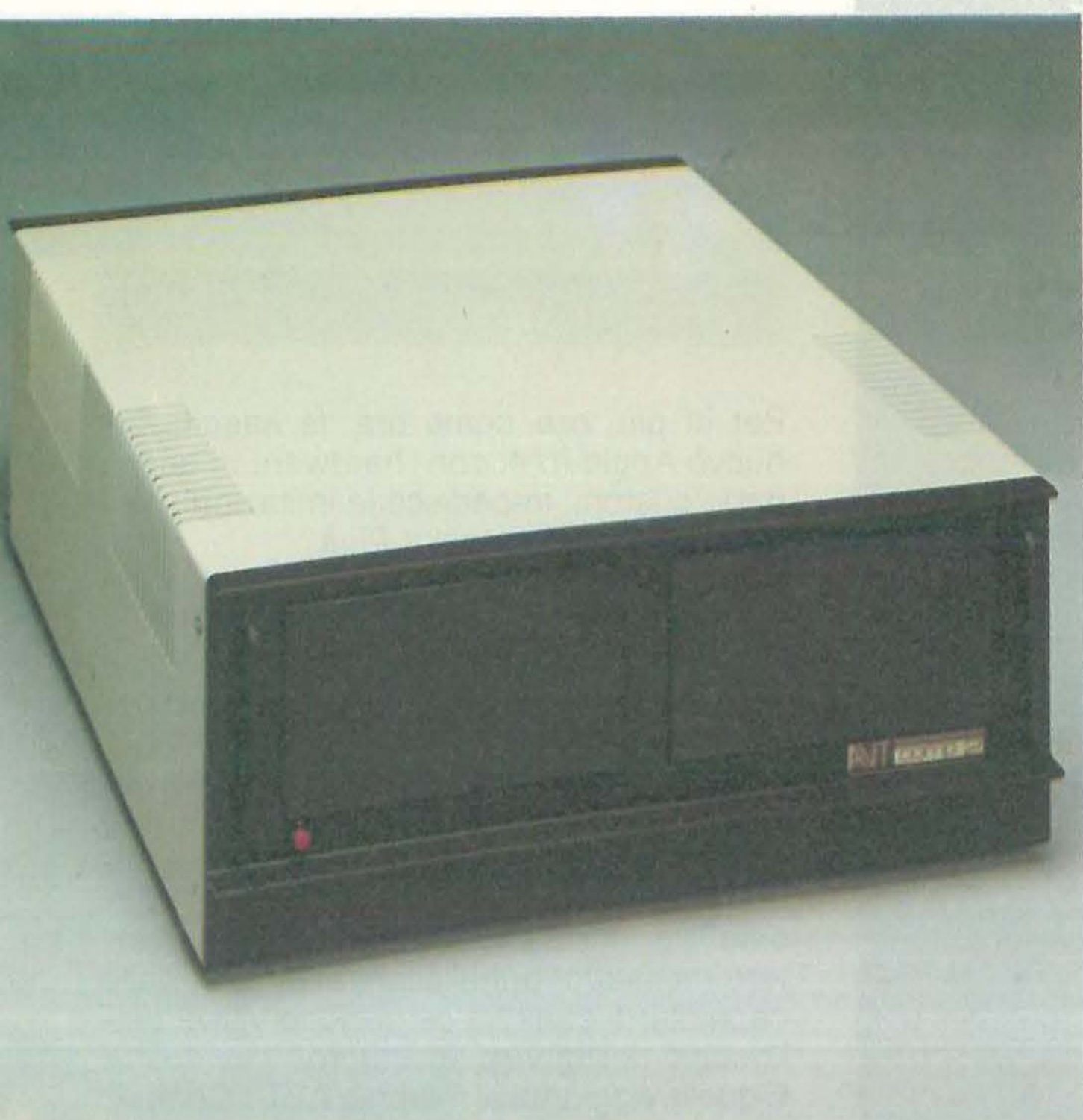
È allora una copia, una imitazione, una banale falsificazione? Nulla di tutto ciò, anzi. Anche perché le recenti dispute legali che in America hanno visto sempre vincente la Apple Computer Inc. portano a supporre che gli imitatori pedissequi siano comunque parecchio malvisti e non possano sperare in molta fortuna.

Il monitor AVT nella versione 12" a fosfori verdi. Sono disponibili anche altre versioni di differenti dimensioni e colori dei fosfori. Lo schermo antiriflesso assicura una piena leggibilità in qualunque situazione.

Il monitor dell'AVT, con aperto lo sportellino che permette di accedere alle regolazioni di luminosità e contrasto, nonché di sincronismo verticale e orizzontale. Sulla destra l'interruttore di accensione e la relativa spia a LED.

Per di più, ora come ora, la nascita del nuovo Apple II/e, con l'hardware in buona parte custom, impedisce le imitazioni, create sulla falsariga del II Plus.
Molto, nell'AVT, richiama alla tecnica del computer delle mele. Nella stessa definizione, non si riscontrano forse analogie fonetiche ben conosciute, soprattutto ponendo attenzione alle prime quattro lettere? Certo, saremmo al limite del condizionamento inconscio. Lo stesso, dicono, che portava una nota multinazionale produttrice di bibite a introdurre fotogrammi "fantasma" nei film di successo, così da trasmettere messaggi pubblicitari, al di fuori delle capacità di conscia percezione visiva dell'occhio dell'ignaro spettatore.
E quella aggiunta al marchio AVT? Comp 2. A voi la scelta: compatibile con un 2 - o II -, oppure semplicemente computer con doppio drive incorporato?
D'altronde le caratteristiche pongono in evidenza la progettazione originale. 64 Kbyte di RAM espandibile a 1 Mbyte non sono certo piccole quantità di memoria e non possono essere confrontate al collega citato.
La tastiera separata, il disegno del contenitore che racchiude il tutto, il monitor: nulla di tutto ciò farebbe supporre a prima vista una compatibilità con alcunché.
Ecco quindi la necessità di un test più approfondito, o perlomeno di una presentazione più completa.

Vista d'insieme

La tendenza riscontrabile nei personal computer di nuova fattura rispecchia la volontà primaria di renderli appetibili per un mercato professionale, oltreché di puro hobby.
Tutte le grandi marche hanno puntato su forme e capacità che richiamano a funzionalità da mini, piuttosto che non da classico home computer.
Anche per la AVT il richiamo di un mercato in effettiva rapidissima espansione, quello dei professionisti più o meno "free" o delle piccole aziende assetate di razionalizzazione, ha consigliato la realizzazione di una macchina di aspetto robusto.
Tre contenitori di differenti forme e dimensioni sono tutto quanto si presenta all'utente di questo AVT Comp 2.
La distribuzione delle funzioni è classica e necessariamente ovvia: un monitor a 12

Sul frontalino dell'unità centrale sono visibili due alloggiamenti coperti con degli sportellini di plastica, pronti a ricevere due drive per floppy da 5" Apple compatibili.

Il retro dell'unità centrale. Oltre al cavo di alimentazione e all'interruttore di accensione, si vedono i connettori per la tastiera e per il video. Cinque feritoie permettono di collegare le periferiche con le schede di interfaccia alloggiabili internamente. Le quattro viti facilmente asportabili rendono accessibili i sette slot di espansione e di interfacciamento, capaci di alloggiare differenti schede Apple compatibili.

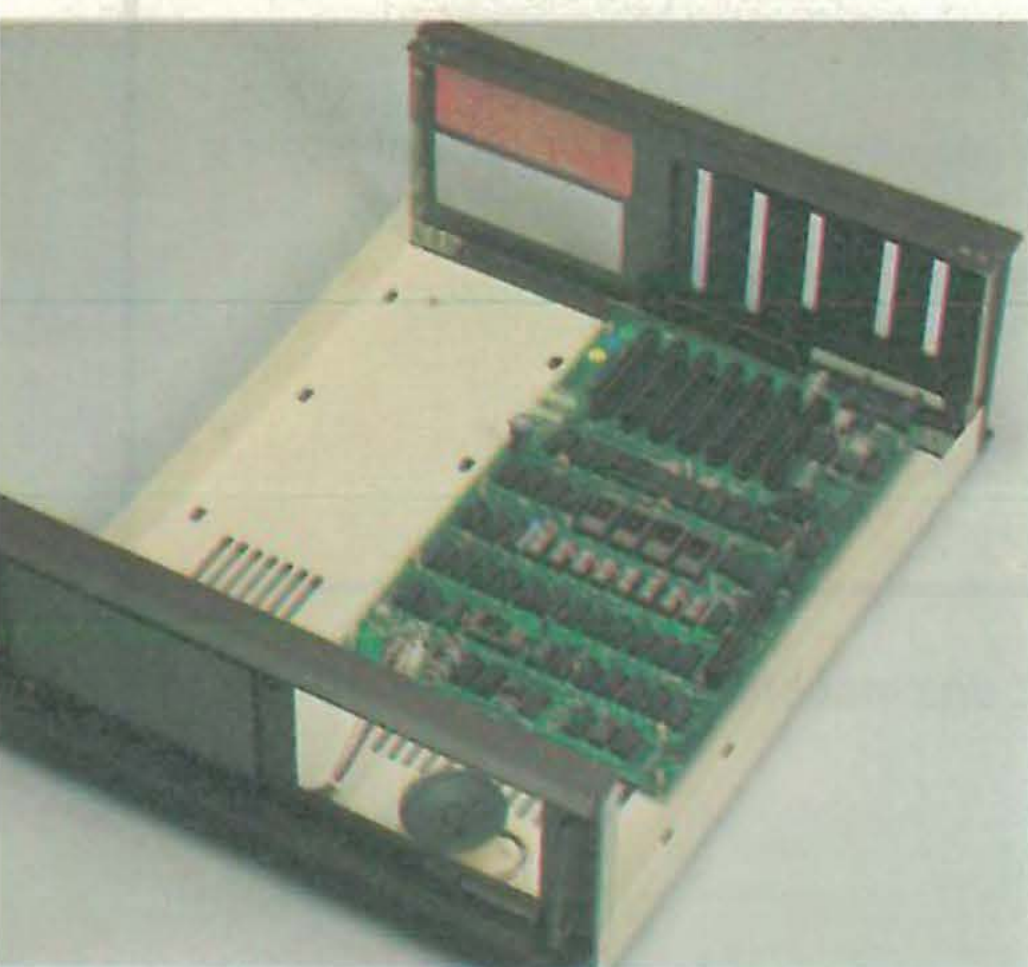
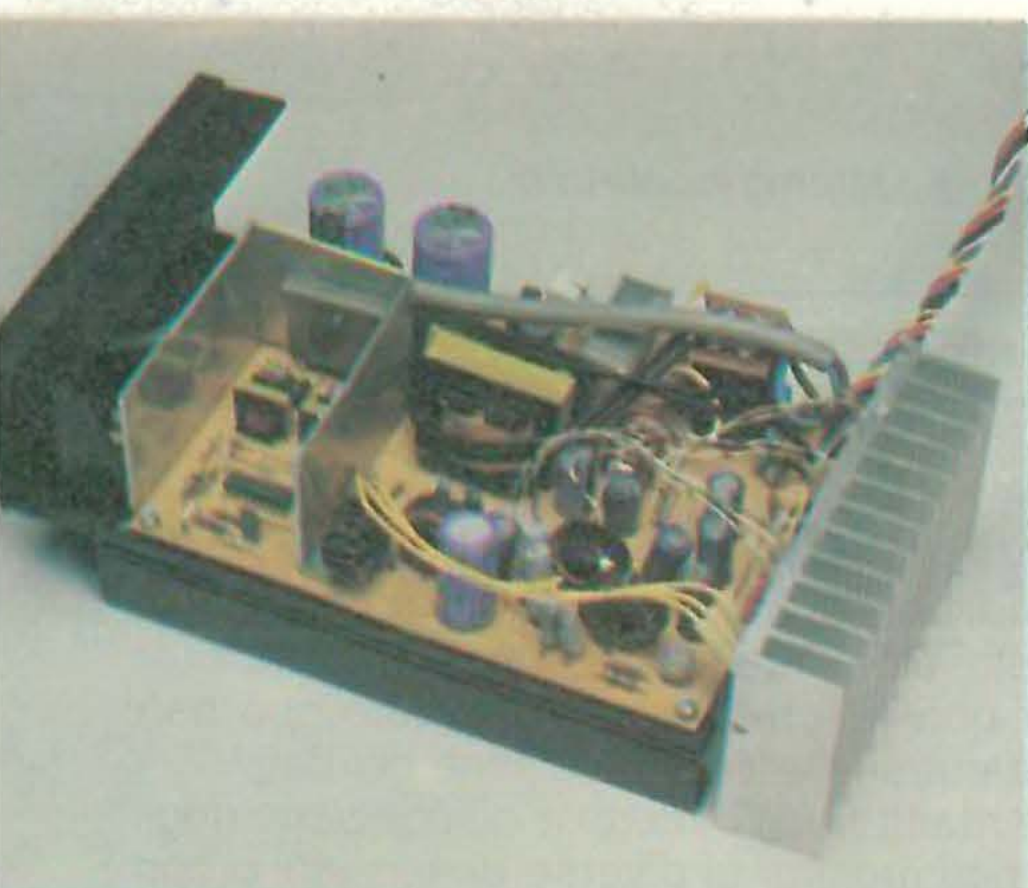
pollici, una tastiera dal basso profilo e l'unità centrale comprende i due alloggiamenti per disk drive da 5".
L'insieme ha invece un aspetto non del tutto ovvio. L'unità centrale è certamente il contenitore di dimensioni maggiori, ma per come è collegato con le diverse sezioni può essere facilmente mascherato in ambienti differenti. L'ufficio o lo studio lo vedranno presumibilmente inserito in un mobile apposito con la sola mascherina frontale resa accessibile per l'inserimento dei floppy. Due profili metallici a maniglia che sporgono leggermente da questa stessa paretina lo rendono adatto anche ad una sistemazione che tende ad assomigliare molto ai tipici rack utilizzati in ambito industriale. L'aspetto gli permette quindi di inserirsi bene anche in ambienti che potremmo definire di produzione, adibito al controllo industriale vicino a più classici strumenti di misura e rilevamento parametri.
E il software per tutto questo? Sanno bene i lettori, ma ormai dovrebbe essere patrimonio comune ai più, che un hardware dalle mille possibilità e di progettazione esemplare rimane un miscuglio di inanimati pezzetti di silicio fino a che non gli viene data l'essenza vitale: il software adatto all'applicazione desiderata.
Ed ecco inserirsi a questo livello la compatibilità citata. Un sistema operativo perfettamente compatibile con quello della casa americana dei personal più diffusi al mondo, rende adeguato l'AVT a compiere le mansioni più disparate. Il software creato per Apple II è, probabilmente, più diffuso e vario di qualunque altro software creato per tutti gli altri personal esistenti. Le applicazioni spaziano dal gioco educativo per bambini ancora sotto l'età della ragione, al controllo di schede apposite che permettono la comunicazione con strumenti di misura o l'applicazione di potenti data base ad uso gestionale. Come proseguiamo a descrivere in dettaglio, AVT è costruito per supportare perfettamente tutto questo, eventualmente completato di tutto l'hardware aggiuntivo anch'esso abbondantemente presente sul mercato.

*Il retro del monitor presenta un ingresso e una uscita per cavetti coassiali a 75 Ohm. È così possibile collegare in cascata più monitor AVT. Un apposito interruttore permette di applicare un carico per terminare la linea.
Si noti, in alto, la robusta maniglia ricavata da un apposito incavo del contenitore stesso. Il trasporto viene così notevolmente facilitato.*

Il modulo principale

Le dimensioni di questa scatola interamente metallica sono ragguardevoli: 38 x 49,7 x 15,7 cm. Il peso non è eccessivo, benché renda il computer un poco scomodo da

L'interno della tastiera mostra la sua provenienza Cherry. Sono visibili il chip multiplexer e la CPU che controlla l'invio delle informazioni al computer ospitante.

65 tasti per un ergonomico basso profilo. Un cavo a spirale notevolmente comodo permette di collegare, tramite un robusto e affidabile connettore a scatto, il computer a questa tastiera. I colori della tastiera sono ovviamente intonati con il rimanente sistema. Dei LED inseriti nei tasti shift e alpha-lock danno un tocco di eleganza all'insieme durante il normale funzionamento.

L'interno dell'AVT dopo aver asportato l'alimentatore switching. Si vede chiaramente la disposizione della piastra del single board computer e, attraverso lo sportellino di un drive per i mini floppy, l'altoparlantino collegato alla piastra stessa.
In fondo gli slot di I/O che permettono di alloggiare qualunque scheda Apple compatibile.

L'alimentatore switching nella sua complessità. È un ottimo alimentatore che fornisce differenti tensioni e correnti mantenendo un alto rendimento.

trasportare: 10,2 kg. Il colore è piacevolmente studiato in abbinamento ad un marrone scuro delle mascherine frontali e posteriori.
Quattro appoggi in gomma assicuranc un posizionamento senza danni anche in mobili o scrivanie di un certo pregio.
Di fronte appaiono subito evidenti i due driver per floppy disk da cinque pollici, con relative fessure di introduzione dei dischi, sportellini di chiusura e LED di segnalazione operatività del disco.
Centrale a queste unità, ben visibile e simpaticamente posizionato, un ulteriore LED rosso segnala l'avvenuta accensione del sistema.
Ancora, nello spigolo a sinistra in basso sporge un piccolo pulsante rosso di reset di sistema. In realtà una duplicazione del testo di break presente sulla tastiera.
Internamente a questo contenitore sono presenti tutte le funzioni essenziali dell'elaboratore.
Un'unica scheda a circuito stampato comprende CPU, memoria, funzioni di I/O, slot di espansione. È una scheda di dimensioni notevoli, posizionata sul fondo e occupante circa metà della superficie di base del contenitore stesso.
Separatamente è sistemato l'alimentatore che fornisce 5 Amper su 5 Volt positivi e 1 Ampere su 5 Volt negativi, nonché + 12V a 1,5 A e − 12V/1 Amp. È racchiuso in una scatola metallica grigliata, per disperdere un calore che comunque è minimo, essendo l'alimentatore stesso di tipo switching e quindi con caratteristiche di alta efficienza.
Un piccolo traliccio metallico, avvitato saldamente alla struttura esterna, sostiene le meccaniche dei floppy disk.
Anche i drive per dischi devono mantenere la compatibilità citata con il personal computer Apple II.
Per ottenere questa caratteristica, l'AVT viene venduto con dei drive di ottima fattura e dalle qualità notevolmente affidabili. Sono degli apparecchi Micro-Sci, una divisione della Standun Controls Inc., californiana.
Questo connubio assicura così continuamente la capacità di supportare in toto il DOS creato dalla Apple, e quindi tutto il software, dai giochi alle applicazioni gestionali, che è possibile trovare su dischetti a 5" in formato Apple.
Lo schedone di base, che realizza un vero single board computer, comprende l'onnipresente microprocessore, in questo caso di tipo 6502, il classico della categoria più tipicamente personal dei computer presenti sul mercato.
Questo gestisce l'intero sistema sotto le

informazioni contenute in una ROM da 16 Kbyte contenente il sistema operativo costituito da monitor e da interprete per BASIC compatibile con Applesoft. La RAM utente è costituita da 8 chip da 64 Kbyte x 1 ciascuna, rendendo immediatamente più capace questo personal rispetto a quelli normalmente presenti sul mercato e adibiti al solo uso degli appassionati.
Il lato di fondo della scheda è quasi interamente occupato da diversi connettori per l'inserimento di quanto necessita al dialogo con il mondo esterno.
Ritroviamo così i tipici slot di I/O che permettono l'alloggiamento delle schede opzionali, per un numero di 7. In un apposito, ulteriore connettore è possibile inserire la scheda per la generazione dei segnali video specifici per il colore. PAL o RGB non sono più una difficoltà per chi desidera rendere colorate le proprie videate.
Due uscite sono collocate sempre nella zona posteriore per il collegamento dell'unità con la tastiera e il monitor. Il cavo proveniente dalla tastiera si innesta facilmente in un connettore ad aggancio automatico che assicura un ottimo collegamento elettrico e meccanico, un soggetto a incontrollabili e disastrose perdite di continuità elettrica. L'altro connettore si adatta facilmente ad un qualunque cavetto schermato di collegamento con il video.
Non sono presenti altri ingressi o altre possibilità di collegamento, se non tramite l'appropriato uso di schede aggiuntive da inserire negli slot citati.
Se in tal senso potrebbe meravigliare l'assenza di una connessione predisposta per l'interfacciamento con un registratore a cassette, bisogna pensare che un sistema di tali caratteristiche si rivolge senzaltro ad un uso professionale o comunque efficiente. Quindi è giusto prevedere l'uso di soli dischi che, d'altronde, hanno raggiunto costi decisamente irrisori rispetto ai vantaggi che rendono disponibili.
Sul retro del contenitore è fissato uno sportellino metallico con tante fessure verticali per far passare i cavi di eventuali schede aggiuntive. È una piastra metallica completamente asportabile ed è l'unico modo agevole di raggiungere l'interno. Viene infatti fissata tramite quattro viti appositamente studiate per essere tolte anche a mano. L'asportazione della piastra suddetta rende agibili gli slot di espansione, benché a volte possa essere un po' scomodo inserire le schede necessarie da questo unico pertugio.
Il layout dei componenti sulla piastra madre differisce in molte parti, come è naturale data l'ampia riprogettazione che è stata operata, rispetto al consimile Apple II.
Anche in tal caso, comunque, vengono rispettati quei criteri di affidabilità che hanno caratterizzato la fortuna dell'Apple. L'assenza di ventole di raffreddamento dimostra che la dissipazione è comunque sotto limiti di pericolosità per i componenti, caratteristica dovuta anche al fatto che le eventuali schede aggiuntive vengono alimentate solamente durante l'effettivo funzionamento.
Tutti gli integrati sono montati su zoccoli, rendendo più agevole una eventuale sostituzione, che non dovrebbe comunque verificarsi neppure nel primo periodo di funzionamento.
Ogni piastra viene infatti testata alla fabbrica e collaudata tramite procedure in burn-in che la rendono insensibile alla cosiddetta mortalità infantile.
Sotto l'alloggiamento dei dischi è pure presente un piccolo altoparlante che poggia su un cuscinetto di gommapiuma. Come per l'Apple il suono di questo trasduttore viene pilotato da opportuni richiami a routine del sistema operativo.

SPECIFICHE TECNICHE

- CPU 6502.
- RAM utente di 64 Kbyte espandibile con schede da 256 Kbyte fino a 1 Mbyte (4 schede da 256 Kbyte).
- ROM di 16 Kbyte con il sistema operativo.
- Uscita video B/N composita.
- Scheda generatrice di segnale a colori in PAL o in RGB.
- Tastiera ASCII separata con 65 tasti, maiuscole, minuscole e tasti cursore.
- 7 slot di I/O per l'alloggiamento di schede opzionali pienamente compatibili con le specifiche Apple - Slot addizionale specifico per la generazione del segnale a colori e per variazioni al tipo di visualizzazione standard.
- Visualizzazione di 40 caratteri x 24 linee.
- Grafica in B/N di 280 x 192 punti o 280 x 160 punti più 4 linee di testo.
- Grafica a 16 colori di 40 x 48 punti o 40 x 40 punti con 4 linee di testo.
- Grafica a 6 colori di 280 x 192 punti o 280 x 160 punti con 4 linee di testo.
- Voltaggio di ingresso 200-240 V.
- Consumo: 80 W.
- Voltaggi interni a pieno carico:
  + 5 V 5 Amp. – 5 V 1 Amp. + 12 V 1,5 Amp. – 12 V 1 Amp.
- Dimensioni:
  CPU 380 x 497 x 157 mm.
  Tastiera 380 x 180 x 15 frontale x 38 retro
- Peso: CPU 10,2 kg. - Tastiera 0,8 kg.

PERIFERICHE OPZIONALI PER L'AVT COMP 2

- Floppy disk con controller e scheda di interfaccia.
- Scheda di interfaccia parallela per stampanti.
- Scheda per la generazione di uno schermo di 80 caratteri x 24 linee con maiuscole e minuscole.
- Scheda di interfaccia per penna ottica.
- Schede di linguaggi per poter utilizzare FORTRAN, COBOL, Pascal, ecc.
- Word processing.
- Paddle per giochi.
- Scheda per interfaccia cassetta e giochi.

Alcune caratteristiche salienti dell'AVT Comp 2, una macchina compatibile con tutto il software creato per Apple II Plus.

Il monitor

Un monitor che ben si adatta al personal AVT, diremmo per consanguineità, è proprio un monitor AVT. In sé il personal è autosufficiente, per cui qualunque monitor di qualità presente sul mercato può ben adattarsi alla visualizzazione dei dati.
In realtà la AVT mette a disposizione degli ottimi monitor di diverse dimensioni e con fosfori variamente colorati. È possibile scegliere, secondo il gusto personale o le considerazioni legate all'igiene visiva, tra fosfori bianchi, verdi o arancioni. È altresì possi-

Programmi di test		Tempi di esecuzione
Ciclo FOR...NEXT	10 FOR I = 1 TO 10000 20 NEXT	*10,78 sec.*
Addizione intera	10 A=2 : B=3 20 FOR I=1 TO 1000 30 C=A+B : NEXT	*3,25 sec.*
Addizione frazionaria	10 A=3.1416 : B=1.4142 20 FOR I=1 TO 1000 30 C=A+B : NEXT	*3,34 sec.*
Moltiplicazione intera	10 A=2 : B=3 20 FOR I=TO 1000 30 C=A★B : NEXT	*4,21 sec.*
Moltiplicazione frazionaria	10 A=3.1416 : B=1.4142 20 FOR I=1 TO 1000 30 C=A★B : NEXT	*5.23 sec.*
Divisione frazionaria	10 A=3.1416 : B=1.4142 20 FOR I=1 TO 1000 30 C=A/B : NEXT	*5,64 sec.*
Radice quadrata	10 A=3.1416 20 FOR I=1 TO 100 30 B=SQR (A) : NEXT	*5,03 sec.*
Logaritmo	10 A=3.1416 20 FOR I=1 TO 100 30 B= LOG (A) : NEXT	*2,43 sec.*
Concatenazione di stringhe	10 A$ =" ": FOR I=1 TO 255 20 A$=A$+"#" 30 NEXT I	*1,54 sec.*
Riempimento di matrice	10 DIM A (20,12) 20 FOR I=1 TO 20 30 FOR J=1 TO 12 40 A (I,J)=3.1416: 50 NEXT J : NEXT I	*5,78 sec.*

bile averli in dimensioni 9" o 12", quest'ultima essendo la diagonale 310 mm che meglio si adatta ad un uso prolungato del sistema.
Sono monitor dall'aspetto robusto e funzionale che presentano diverse regolazioni utili e un aspetto di possibile collegamento in cascata, che risulta interessante.
Inseriti in cubi metallici con una maniglia di presa sul lato superiore, maniglia incassata nel mobile stesso, si presentano con uno schermo antiriflesso che protegge anche dalla polvere. Dietro un piccolo sportellino ricavato sotto lo schermo stesso sono presenti ben quattro regolazioni. Le classiche manopoline che controllano contrasto e luminosità, nonché spostamento orizzontale e sincronismo verticale. A lato dello sportellino un LED rosso incassato nel mobile avverte della corretta alimentazione data premendo un comodo pulsante frontale.
Sul retro un ulteriore interruttore commuta tra funzionamento normale e funzionamento in cascata. Difatti sono presenti due prese coassiali: un ingresso e una uscita. Tramite queste e dei comuni cavetti coassiali da 75 ohm è possibile collegare in cascata fino a 10 monitor AVT. In pratica l'interruttore in questione va posto in posizione bassa nel monitor che termina la catena, così da caricare opportunamente la linea che altrimenti rimarrebbe scollegata.
Tutto ciò risulta essere particolarmente utile in situazioni ove differenti persone debbono osservare lo schermo senza intralciarsi tra loro. Tipicamente in aule didattiche, luoghi che apprezzeranno senz'altro più di altri tali caratteristiche abbinate ad un elaboratore tanto versatile.

La tastiera

Le dimensioni della tastiera sono 380x180x15 frontale x 38 posteriore, indicanti il basso profilo di una struttura leggermente a cuneo, per una migliore ergonomicità.
Tale contenitore supporta una tipica tastiera Cherry a 65 tasti alfanumerici.
È una tastiera comprendente un chip di decodifica della matrice dei pulsanti costituito da un multiplexer da 16 linee a 4 linee, il quale comunica con una CPU 8048 che gestisce la trasmissione verso il micro computer.
Tutte le funzioni sono ottenute tramite pulsanti con codici differenti. I tasti di control, alpha, shift e shift lock sono gestiti direttamente dal 8048, il quale pilota anche i due LED di segnalazione che poi sono inseriti entro i tasti stessi di alpha e di lock.
Il tasto break è invece al di fuori del controllo dei circuiti di tastiera e non fa altro che porre a massa una apposita linea di uscita.
Un quarzo fornisce il clock agli integrati e un cavo trasferisce i dati e i segnali di controllo all'esterno.
È presente anche un cavallotto per cambiare la tastiera in versione tedesca o internazionale.
Tale tastiera è collegabile al personal AVT tramite un comodissimo cavo a spirale, sufficientemente lungo per posizionarla anche piuttosto distante dall'unità centrale.
È così grandemente facilitata la razionale disposizione di ogni singola unità componente il sistema.
I tasti potrebbero essere forse un poco più sensibili al tocco, ma il basso profilo dell'insieme facilita comunque la digitazione.
Rispetto alla tastiera che è nata insieme al sistema operativo adottato anche da questo personal, sono presenti molte funzioni in chiaro. Così come lo spostamento del cursore, o i simboli di sottolineatura ed altri che normalmente sono raggiungibili solo attraverso più laboriose digitazioni di control e altri tasti.
Una importante caratteristica è il repeat automatico, tale per cui è sufficiente tenere premuto il tasto per qualche frazione di secondo in più, che il simbolo ad esso associato viene inviato continuamente all'elaboratore fino al rilascio del tasto stesso.
Sono tutti piccoli particolari che in applicazioni di word processing, per citare un esempio ormai classico, hanno notevole importanza.
Forse l'unica piccola pecca imputabile alla realizzazione di questa tastiera risiede nella leggera tendenza all'incurvamento della scheda a circuito stampato che sostiene i tasti. Un appoggio interno oltre quelli già presenti, possibilmente in zona centrale,

non avrebbero guastato.
La tastiera è l'elemento più leggero dell'insieme AVT: solo 8 etti.

Operatività

Collegato il sistema alla rete a 220 V, saranno necessari dei convertitori per spine da passo tedesco, a passo europeo, e acceso sia il monitor che l'unità centrale, il sistema è subito pronto per essere utilizzato se mancano i dischi, mentre si aspetta l'inserzione del disco contenente il DOS (Disk Operating System) se sono presenti i drive. Ciò significa che il monitor di sistema è di tipo autostart, nel senso che all'accensione si entra immediatamente in ambiente BASIC. Se invece è presente il disco aspetta di caricare il sistema operativo opportuno. Ciò permette di lanciare procedure senza altri comandi se non con l'operazione semplicissima di introduzione del disco stesso. La realizzazione di sistemi semplici da usare viene così garantita per chiunque, ma d'altronde questa era proprio una caratteristica dei sistemi Apple e quindi anche di questo compatibile AVT.
Comunque appare sul monitor la scritta AVT II, che indica il buon funzionamento del tutto.
Apparsa la parentesi quadra chiusa, seguita dal cursore lampeggiante, possiamo digitare i comandi BASIC Applesoft o quelli relativi ai dischi, se questi sono presenti.
Le funzioni di editing sono tipiche Apple. Le due frecce presenti sulla tastiera sono utili per cancellare o ricopiare in memoria quanto si va scrivendo sul monitor, mentre sarà sempre necessario spostarsi con uno dei tasti *I*, *K*, *M* e *J*, che sulla tastiera formano il classico diamante, preceduti da *ESC* se si vuole un puro e semplice spostamento di cursore. In effetti con AVT viene venduto un dischetto DOS apposito, ma per ora è evidentemente una semplice copia del DOS Apple, giacché si potrebbero meglio sfruttare i codici generati dalla tastiera più completa per un vero e più comodo indirizzamento del cursore.
A differenza dell'Apple, esteso a 64 Kbyte, quando si carica da disco il DOS su AVT non viene automaticamente richiamato l'Integer BASIC. È comunque possibile averlo in memoria, giacché questa è normalmente da 64 Kbyte, caricandolo in un momento successivo.
Il BASIC Applesoft di questo AVT e studiato per gestire ogni caratteristica disponibile. Sono quindi presenti, ad esempio, tutte le istruzioni che gestiscono anche la grafica, sempre possibile in bassa o alta risoluzione. Ciò significa poter accedere sullo schermo a 280x160 punti con 4 linee di testo, oppure a 280x192 punti in alta risoluzione.
Schede opzionali rendono disponibile anche la visualizzazione a 80 colonne, dedicata ad usi più professionali di WP.

Costruttore:

P.D. BOX 61411
Regentesseplein 231
2506 AK Den-Harg
Holland

Distributore:

Rebit Computer
Via Induno, 18
20092 Cinisello B. (MI)
Tel. 02/6122371

Listino prezzi:

AVT Comp 2 64 Kbyte	*L. 1.499.000+IVA*
Guida all'uso Apple II	*L. 26.000+IVA*
Scheda RGB color	*L. 50.000+IVA*
Disk drive	*L. 750.000+IVA*
Controller per 2 disk drive	*L. 200.000+IVA*
Interfaccia per stampante parallela	*L. 240.000+IVA*
Scheda per video 80 colonne	*L. 440.000+IVA*
Paddle (coppia)	*L. 27.000+IVA*
Monitor 9" fosfori bianchi	*L. 272.000+IVA*
Monitor 9" fosfori verdi	*L. 272.000+IVA*
Monitor 9" fosfori arancioni	*L. 280.000+IVA*
Monitor 12" fosfori verdi	*L. 290.000+IVA*
Monitor 12" fosfori arancioni	*L. 300.000+IVA*
Stampante Honeywell S10	*L. 700.000+IVA*
Stampante Seikosha GP100A	*L. 550.000+IVA*
Stampante Seikosha GP250X	*L. 635.000+IVA*

Le opzioni

Alla Rebit, il rivenditore di AVT, sono attualmente disponibili alcune delle possibili opzioni create per lo stesso. In particolare la scheda RGB per il colore, la scheda di controllo per disk drive che permette di aggiungere ulteriori drive anche oltre i due normalmente acquistati, una interfaccia parallela per stampante, la scheda per il video a 80 colonne, le paddle per utilizzare il personal anche quale strumento frivolo.
Molte altre opzioni si possono scegliere tra le innumerevoli schede create per Apple, anche da case differenti che ne garantiscano la compatibilità.
Si spazia da schede che permettono il controllo di una penna ottica, a quelle di lettura di codici a barre destinati ad usi commerciali e di classificazione, fino a schede che rendono il personal computer uno strumento di misura efficientissimo e dotato di capacità impensabili. Una casa americana rende addirittura disponibile una grossa scheda inseribile negli slot che trasforma il vostro personal in sofisticato oscilloscopio con memoria.

Conclusioni

Avendo un personal così flessibile è sempre molto difficile individuare con precisione un campo di applicazione specifico. Si spazia dalle applicazioni hobbistiche alle applicazioni gestionali di piccole aziende, soprattutto se pensiamo alle infinite possibilità di collegamento con memorie di massa più capaci, quali hard disk di notevoli dimensioni.
La grafica lo rende strumento ideale in campo didattico e di word processing che richieda frequenti realizzazioni di figure e disegni. La versatilità e le possibilità opzionali di questo personal AVT spingerebbero a lavorare di fantasia oltre ogni limite.
Sicuramente, comunque, ci troviamo di fronte ad un personal che ha il grosso pregio di costare poco e di essere sapientemente assistito e seguito da una tra le più importanti società italiane: la Rebit, divisione computer della ben nota ed affidabile GBC. ■

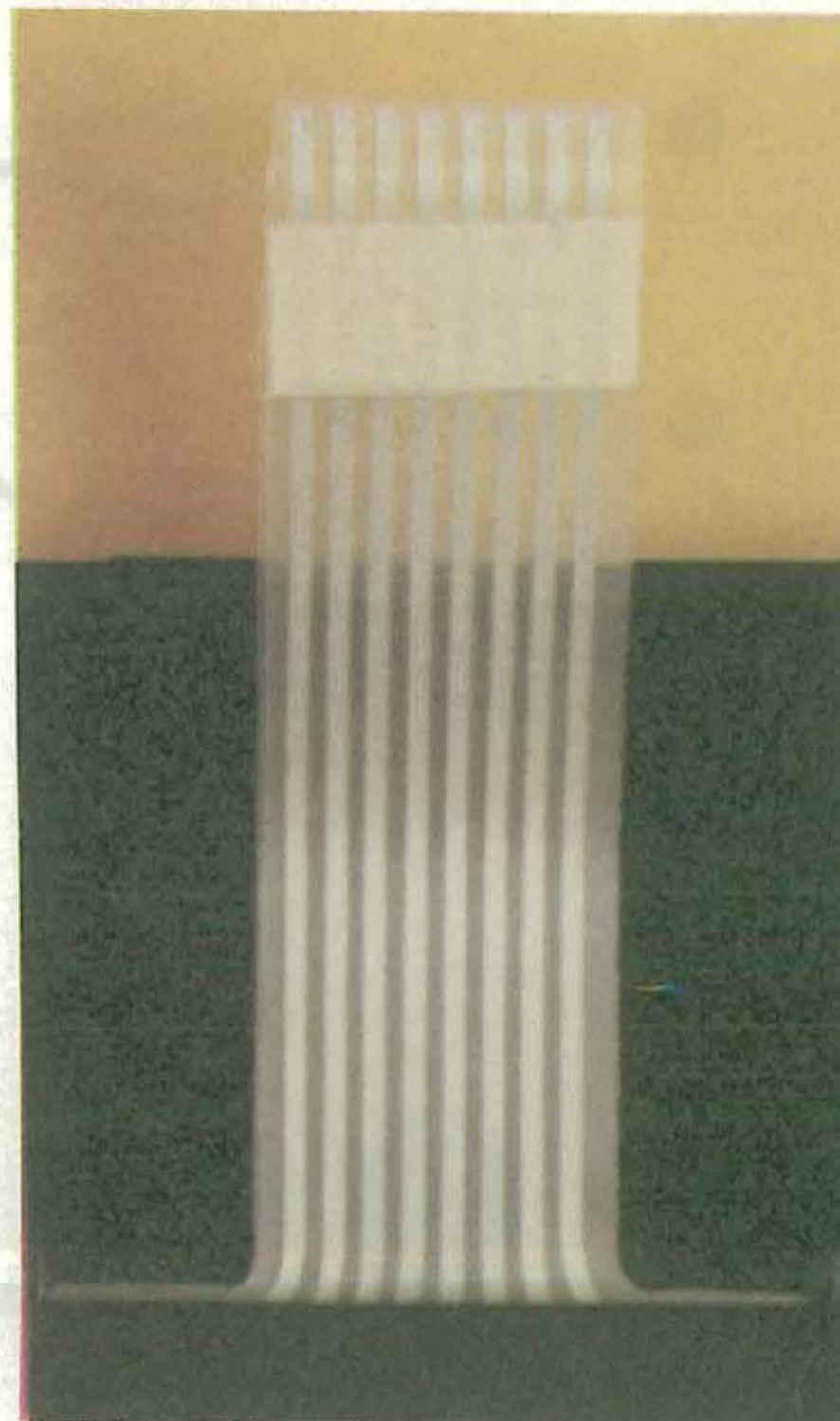
a cura della **Redazione**

In una significativa tendenza al ribasso dei prezzi, il mercato dei calcolatori ha prodotto negli ultimi anni una lunga serie di modelli molto interessanti di cui i Sinclair ZX 80 e ZX 81 rappresentano tuttora i calcolatori completi ridotti ai minimi costi e, grazie all'uso di tecnologie avanzate, anche alla massima semplicità costruttiva. Lo ZX 80 in particolare ha sancito per primo il principio del calcolatore come oggetto domestico di consumo, rendendo l'informatica materia di studio pratica e accessibile a tutti. Sulla scia di un meritato successo, la stessa

L'arcobaleno di colori che caratterizza la tastiera dello Spectrum è una tipica nota stilistica che richiama alle potenti funzioni grafiche di questo personal.

Il retro dello Spectrum. È evidente l'aspetto dimensionale del sistema: compattissimo e sottile, pur mantenendo una comodità d'uso notevole, anche dovuta al fatto che tutte le funzioni sono riportate in chiaro sulla tastiera. Si vedono gli ingressi di alimentazione a 9 Volt e del registratore a cassette, nonchè le uscite per il segnale video adatto ad un qualunque televisore a colori o B/N e ancora per il registratore. È visibile, sulla sinistra, anche il connettore per il bus di sistema.

Sinclair ha presentato lo ZX Spectrum, appena arrivato nei Bit Shop italiani, con un BASIC residente molto potenziato.
Al primo impatto, noi che già conoscevamo lo ZX 81, cui lo Spectrum assomiglia vagamente, causa la stessa plastica nera di buona fattura, siamo rimasti favorevolmente impressionati dall'aspetto del nuovo nato. Per prima cosa se ne notano la larghezza e la profondità, quasi scambiate tra di loro rispetto ai vecchi modelli (cm. 23,3x14,2x3), che lo rendono equilibrato alla vista. La tastiera, che resta fedele al vecchio

Il connettore flessibile che unisce la scheda della tastiera agli appositi zoccoli interni. Realizzata con un flessibilissimo circuito stampato, è indice della moderna tecnologia da cui nasce lo Spectrum.

Bitest: Sinclair ZX Spectrum

schema dei 40 tasti, e non uno di più, ne prende tutta la larghezza e i due terzi della profondità, e, gaudete, è formata da tasti veri, e non più da sensori di dubbio aspetto. Ogni tasto è composto da vera e propria gomma che lo rende non scivoloso e, tutto sommato, piacevole al tatto.

Non si possono non notare le numerose scritte che appaiono su, sopra e sotto ogni tasto; quelli numerici ne hanno ben sette, corrispondenti ad altrettanti possibili impieghi. Gli altri hanno in genere sei funzioni.

Gli ottimi zoccoli di alloggiamento dell'espansione di memoria, che porta il sistema alla ragguardevole capacità di 48 Kbyte di RAM.

Oltre all'*ENTER* e allo spazio, un po' più largo degli altri che ha anche la funzione di *BREAK*, ci sono due *SHIFT* distinti; *CAPS SHIFT*, che serve anche ad ottenere le maiuscole, e *SYMBOL SHIFT*, con la scritta in rosso, che serve ad ottenere i simboli dello stesso colore.

Come gli altri Sinclair, lo Spectrum si può trovare in diversi modi che condizionano l'input: i tasti hanno differenti funzioni a seconda del modo, oltre che degli *SHIFT* premuti. I modi principali sono il modo *K* e il modo *L*, dove *K* stà per keyword (parola chiave), e *L* per lettera. Lo Spectrum accetta le parole chiave con una sola battuta: nel modo *K* un tasto corrisponde alla parola chiave scrittavi sopra in bianco, invece che alla lettera cui corrisponderebbe in modo *L*. I due modi si alternano automaticamente, cioè il calcolatore sa sempre a priori se la sintassi richiede una parola chiave o una lettera, e lo evidenzia scrivendo la sigla del modo all'interno del cursore. Per unificare conseguentemente la sintassi delle linee, nei Sinclair è stato reso obbligatorio il comando *LET*, generalmente facoltativo. Gli altri modi possono essere considerati come appendici dei due modi principali. Il modo *C (C* = caps lock), quando è attivo, sostituisce il modo *L* facendo sì che tutte le lettere battute vengano considerate maiuscole. Il modo *G* permette di accedere agli otto caratteri grafici predefiniti sui tasti numerici, od ai ventuno caratteri grafici definibili dall'utente corrispondenti alle lettere da *A* a *U*. Si dovrebbe aver già capito che lo Spectrum ha un set di caratteri molto potente, diversamente dai suoi predecessori. Questa è la prima di una serie di caratteristiche che a volte desidereremmo avere su macchine di costo e dimensioni ben superiori. Ci rimane da esaminare solo il modo *E (E* = esteso) che permette di ottenere la maggior parte delle funzioni e dei comandi e qualche carattere in più. Si ottiene premendo contemporaneamente i due *SHIFT* e fa sì che i tasti corrispondano, per una sola battuta, alle scritte verdi e rosse sopra e sotto di essi. A seconda di quando si usa può essere un estensione di entrambi i modi principali.

La terza cosa che abbiamo notato sono state quattro strisce diagonali colorate sulla destra dell'apparecchio, che rivelano discretamente la capacita di generare colori, definitivamente confermata dalla colorata lista degli stessi (blu, rosso, magenta, verde, ciano, giallo, bianco, nero) sopra i tasti numerici.

Lo schermo ha normalmente disponibili per la stampa 22 linee di 32 posizioni. Lo schermo ha in realtà 24 linee, ma la 23 e la 24 sono permanentemente riservate per l'input e i messaggi di errore. La gestione del colore avviene tramite una serie di attributi che determinano individualmente per ogni posizione: il colore dello sfondo; il colore della scrittura; se la scrittura debba essere a luminosità doppia; se la posizione debba lam-

La ULA, il circuito custom che permette allo Spectrum potentissime funzioni in spazi decisamente ristretti.

peggiare scambiando continuamente lo sfondo con la carta e viceversa. Per completare l'arcobaleno la cornice dello schermo può tingersi di uno dei colori disponibili.
Per quanto riguarda il video resta da descrivere la grafica ad alta risoluzione che sfrutta i comandi *PLOT, DRAW* e *CIRCLE* per disegnare su una matrice composta da 8x8 pixel per ogni posizione di stampa per un totale di 256x192=49152. *PLOT X,Y* disegna un puntino di coordinate (X,Y), considerando lo schermo come primo quadrante degli assi cartesiani. *DRAW X,Y,A* traccia una linea a partire dalla posizione raggiunta dall'ultimo comando grafico, spostandosi di *X* orizzontalmente, di *Y* verticalmente e girando di *A* radianti; se si omette *A*, traccia una linea retta. *CIRCLE X,Y,R* traccia una circonferenza di centro (X,Y) e di raggio R. Purtroppo la grafica non riesce a sfruttare al massimo i colori dato che in una posizione di carattere di 8x8 pixel non si possono avere più di due colori (scrittura e sfondo). Questo comunque non dovrebbe causare problemi nella maggior parte dei casi; limitazioni di tipo analogo esistono anche su sistemi più grossi.
Gli attributi di cui si è già parlato vengono settati tramite i comandi *INK, PAPER, BORDER, BRIGHT, FLASH* e *INVERSE* (controlla l'inversione della matrice di punti per la stampa). Se questi vengono usati da soli stabiliscono gli attributi permanenti; se invece sono usati all'interno di un comando *PRINT* o *INPUT*, stabiliscono degli attributi temporanei, che prevaricano i permanenti solo per il comando che li contiene. Il controllo del colore può essere ottenuto anche tramite gli appositi caratteri di controllo, che sono efficaci ovunque inseriti, rendendo anche possibile "colorare" un programma, facendo sì che il suo listato appaia automaticamente in sezioni di diversi colori.
Notevoli sono anche i comandi *AT* e *TAB*, utilizzabili all'interno di una *PRINT* e di una *INPUT*. *AT R,C* sposta la posizione di stampa alla riga *R*, colonna *C*. *TAB n* stampa abbastanza spazi da portare la posizione di stampa alla colonna *n*. Se *n* è maggiore di 32, viene ridotto modulo 32.
Ultima cosa relativa allo schermo è il modo di trattare i dati in input che sfrutta la sezione bassa dello schermo come buffer.

Il banco di memoria da 16 Kbyte, ottenuto con le classiche 4116. Pur non essendo montate su zoccolo, l'ottimo layout permette una rapida assistenza su un circuito comunque affidabilissimo.

Il beeper ottenuto con un altoparlante ben collocato in uno spigolo della piastra madre.

Il compattissimo interno contiene la CPU Z80, il circuito ULA di progettazione Sinclair che svolge quasi tutte le funzioni di sistema, la ROM contenente il software del monitor e dell'interprete, la memoria RAM. In alto a sinistra il modulatore, schermato, per la generazione dei segnali video.

In effetti, quando si introduce una linea di programma o si risponde ad un comando di input si vede apparire tutto ciò che viene battuto in detta sezione, donde viene rimosso non appena si preme *ENTER*. La sezione in questione, comunque, si può espandere fino ad occupare tutto il video e, poichè si possono usare i comandi *AT* e *TAB* anche in un comando di input, è possibile creare facilmente tabelle per un input complesso in forma chiara, cosa, anche questa, di cui abbiamo sentito la mancanza su altre macchine.
Per l'input dei dati si può usare anche la funzione *INKEY$* che ritorna il tasto premuto al momento della sua chiamata, o una stringa nulla se non vi è alcun tasto premuto. Un'altra funzione che sente se viene premuto un tasto è *PAUSE n* che sospende l'esecuzione del programma per *n* cinquantesimi di secondo, o per sempre se *n* è uguale a zero. Fortunatamente una pausa così ottenuta può essere troncata premendo un tasto dopo il suo inizio. *PAUSE 0* risolve in un solo comando il problema di aspettare che l'operatore decida di continuare l'esecuzione del programma, magari dopo avere letto qualcosa dal video.
La correzione delle linee di programma avviene in modo esattamente identico ai vecchi modelli: ci si porta col cursore di programma (un simbolo > tra il numero di linea e il primo comando che indica l'ultima linea introdotta) sulla linea che si vuole correggere, si preme *EDIT*, che fa sì che detta linea venga riportata sulla parte bassa dello schermo, ci si sposta col cursore dopo gli eventuali caratteri da cancellare, si cancellano con *DELETE*, si introducono i nuovi e si preme *ENTER* per finire.
Anche l'indicazione degli errori di sintassi non è cambiata. Lo Spectrum non accetta linee contenenti errori e posiziona un ? lampeggiante dopo il carattere che ha causato l'errore, rifiutandosi di proseguire se non viene corretto; questo sistema ne facilita l'identificazione. Ci sono poi 25 diversi messaggi di errore per indicare qualcosa che non va durante l'esecuzione di un programma o di un comando.

Lo Spectrum, in Italia, nasce già con tutto il bagaglio software presente nel mondo. Cassette di programmi già ottimizzati e garantiti sono disponibili presso i rivenditori, e molto software comincia a circolare tra gli appassionati.
Il tutto viene venduto con ottimi manuali che guidano l'utente ad un corretto e semplice uso del sistema.

Programmi di test		Tempi di esecuzione
Ciclo FOR...NEXT	10 FOR I = 1 TO 10000 20 NEXT	*44,3 sec.*
Addizione intera	10 A=2 : B=3 20 FOR I=1 TO 1000 30 C=A+B : NEXT	*8,8 sec.*
Addizione frazionaria	10 A=3.1416 : B=1.4142 20 FOR I=1 TO 1000 30 C=A+B : NEXT	*9,16 sec.*
Moltiplicazione intera	10 A=2 : B=3 20 FOR I=TO 1000 30 C=A★B : NEXT	*9,16 sec.*
Moltiplicazione frazionaria	10 A=3.1416 : B=1.4142 20 FOR I=1 TO 1000 30 C=A★B : NEXT	*10,26 sec.*
Divisione frazionaria	10 A=3.1416 : B=1.4142 20 FOR I=1 TO 1000 30 C=A/B : NEXT	*10,73 sec.*
Radice quadrata	10 A=3.1416 20 FOR I=1 TO 100 30 B=SQR (A) : NEXT	*12,80 sec.*
Logaritmo	10 A=3.1416 20 FOR I=1 TO 100 30 B= LOG (A) : NEXT	*8,20 sec.*
Concatenazione di stringhe	10 A$ =" ": FOR I=1 TO 255 20 A$=A$+"#" 30 NEXT I	*4,70 sec.*
Riempimento di matrice	10 DIM A (20,12) 20 FOR I=1 TO 20 30 FOR J=1 TO 12 40 A (I,J)=3.1416: 50 NEXT J : NEXT I	*3,23 sec.*

Tabella comparativa dei tempi.

C'è qualcosa di interessante da dire anche a proposito di come lo Spectrum tratta le stringhe, o meglio di come le divide in modo non standard. Le variabili stringa sono normali, di lunghezza variabile, a differenza delle matrici di stringhe, come si vedrà. Chi è già familiare con il BASIC non può non conoscere le funzioni stringa *LEFT$*, *RIGHT$*, *MID$*, *TL$* che servono per estrarre una sottostringa da una stringa data, e forse avrà avuto dei problemi per ricordarsi se il tal carattere era compreso o meno nella tal suddivisione. Ebbene tutte queste sono state sostituite, e vorremmo dire genialmente, con la notazione unica (*M TO N*) che ha ovviamente senso solo dopo un espressione stringa e che ritorna detta espressione dal carattere emmesimo all'ennesimo compresi. In più, se *m* viene omesso viene assunto uguale ad 1, mentre *n* omesso viene assunto uguale alla posizione dell'ultimo carattere dell'espressione data; se si vuole un solo carattere della stringa origine è sufficiente racchiudere la sua posizione tra parentesi. Un'espressione stringa che contenga anche tale notazione ha, ovviamente, lunghezza predefinita, ma la cosa più interessante è che può essere assegnata. In altre parole se si vuole far sì che il quarto, il quinto e il sesto carattere della stringa *A$* diventino *"ABC"* è sufficiente scrivere: *LET A$ (4 TO 6) = "ABC"*. In questi tipi di assegnazioni a lunghezza fissa lo Spectrum tiene come riferimento il "margine" sinistro e colma con spazi o tronca a destra l'eccedenza.
Anche le stringhe di una matrice di stringhe sono di lunghezza prestabilita, specificata come ultima dimensione nella relativa *DIM*. In effetti lo Spectrum ha più che altro una gestione molto sofisticata di matrici di singoli caratteri, grazie alla quale, omettendo l'ultimo indice in un comando di assegnazione, si accede a tutta la stringa, di lunghezza fissa, dei caratteri che hanno gli indici specificati in comune; specificando anche l'ultimo indice si ottiene, ovviamente, il relativo carattere della stringa in questione. Al suo posto si può usare anche la notazione *TO* che, come già visto, consente di estrarre delle sottostringhe.
Presenta un certo interesse anche il modo in cui si possono leggere le stringhe da tastiera. Oltre alla *INKEY$* si possono usare la *INPUT* e la *INPUT LINE*. La *INPUT*, come sugli altri Sinclair, ma diversamente da molti altri BASIC, stampa le virgolette intorno al cursore. Quando si preme *ENTER* vengono lette dal buffer di input insieme agli altri caratteri e fanno sì che la routine di lettura capisca che ha proprio a che fare con una stringa. L'utilità di questo sistema è che le virgolette possono essere cancellate per dare al calcolatore un'espressione stringa da cui ricavare il valore stringa richiesto (per esempio *VAL$*). *INPUT LINE* si ottiene posizionando il suffisso *LINE* ovunque in un comando *PRINT*, e non come parola chiave a sè stante. Fa sì che il calcolatore non stampi le virgolette, ma si comporti come se ci fossero, non consentendo più di cancellarle. Si comporta quindi come la *INPUT* standard BASIC, con un'estetica, a nostro parere, preferibile. Si noti anche in questo caso l'estrema flessibilità del comando *INPUT*: in una sola linea è possibile leggere variabili con e senza l'opzione *LINE*.
Non ci sono problemi neanche con i nomi delle variabili numeriche semplici. Va bene una qualunque sequenza, lunga a piacere, di caratteri alfabetici o numerici, anche contenente spazi, con la sola condizione che il primo sia una lettera. Gli spazi e il fatto che le lettere siano maiuscole o minuscole non contano per distinguere tra loro le variabili; così, per esempio, *"Il gattino nero"* e *"ILGATTINONERO"* sono entrambi nomi validi che si riferiscono allo stesso valore numerico. Si consideri che su molti sistemi molto costosi i nomi di variabili sono soggetti a severe restrizioni.
Purtroppo i nomi delle variabili stringa, delle variabili con indice e delle variabili di controllo dei cicli *FOR ... NEXT* sono ristretti a lettere singole. Per le variabili stringa non possono neanche coesistere in memoria una matrice e una variabile semplice dello stesso nome, cosa invece ammessa con le variabili numeriche. Non riteniamo comunque che quest'ultimo fatto abbia molta importanza; più grave è, invece, la restrizione sui nomi delle variabili stringa.
Lo ZX Spectrum ha anche tendenze musicali, infatti è dotato del comando *BEEP D,A* che fa emettere dal trasduttore incorporato una frequenza della durata di *d* secondi e di altezza a, dove a, positivo o negativo anche frazionario, indica la distanza in semitoni dal DO centrale normalizzato (261,6 Hz). La gamma di frequenze coperta è molto ampia anche se la bontà del suono si mantiene solo per

quattro o cinque ottave. Chi non fosse soddisfatto della fedeltà o del volume resi, può facilmente collegarsi ad un qualunque amplificatore esterno, infatti il segnale audio è riportato sulle prese per il registratore poste sul retro dell'apparecchio. E' analogamente possibile collegarsi a circuiti più sofisticati per generare una più vasta gamma di suoni.
Anche la faccia logica del calcolatore desta interesse. Accanto all'usato *IF ... THEN* limitato forse dall'assenza dell'*ELSE*, gli operatori logici *AND, OR* e *NOT* permettono, anche con l'ausilio delle parentesi, di concatenare a piacere condizioni logiche diverse. Più interessante è l'uso che se ne può fare con una variabile normale e una variabile logica. *X AND Y* ha il valore di *X* se *Y* è vero, mentre è uguale a zero se *Y* è falso; *X OR Y* ha valore 1 nel primo caso, valore *X* nel secondo. Poiche *X* può essere anche stringa e vero corrisponde a 1 (non -1 come standard) e falso a zero, sono possibili linee tipo: *PRINT (A AND A> = B)+(B AND A<B)* che stampa sempre il più grande tra *A* e *B*, o: *PRINT A$;("<" AND A$<B$)+ ("=" AND A$=B$)+(">" AND A$>B$);B$* che stampa la giusta relazione tra *A$* e *B$*, piu comodamente di una serie di *IF*.
Il BASIC ha, per il resto, tutte le funzione che dovrebbe avere. Sono disponibili *READ, DATA* e *RESTORE*, le funzioni *LEN, STR$, VAL, VAL$, SGN, ABS, INT, SQR, FN, PI, EXP, LN, SIN, COS, TAN, ASN, ACS, ATN*, è possibile definire funzioni personalizzate con *DEF FN*. In più *RND* ritorna un numero pseudo casuale tra 0 e 1 escluso, mentre *RANDOMIZE* cambia il seme usato dalla prima.
Per arrivare più nel cuore del sistema ci sono la funzione *PEEK* e il comando *POKE* che permettono rispettivamente di leggere e scrivere direttamente nelle locazioni di memoria. A proposito di memoria lo ZX Spectrum viene fornito con 16 Kbyte di RAM, espandibili a 48 Kbyte e BASIC residente a parte su ROM, mentre non accetta le vecchie espansioni 3 Kbyte e 16 Kbyte dei modelli precedenti. Il comando *CLEAR* permette di spostare, per riservare dello spazio, il puntatore che delimita la zona RAM utilizzabile dal BASIC (per esempio per una routine in Assembler). Oltre a ciò *CLEAR* pulisce lo schermo (*CLS*) e cancella tutte le variabili. *USR* può essere usata con argomento numerico per chiamare una routine Assembler all'indirizzo specificato, e ritorna il valore della coppia di registri *BC*. In modo insolito con una lettera per argomento ritorna l'indirizzo della maschera di punti per il carattere grafico definibile dall'utente corrispondente alla lettera specificata.
I numeri vengono conservati in memoria in 5 byte in formato floating point con esponente compreso tra −38 e +38 e con circa nove cifre significative e mezzo di cui la *PRINT* ne stampa otto. Il numero intero più grande che si possa memorizzare con assoluta precisione è 4294967295. I numeri interi compresi tra −65535 e +65535 vengono memorizzati in notazione normale in 2 byte + 1 per il segno, ma occupano sempre 5 byte sprecandone due, cosa invero alquanto originale.

Costruttore:
Sinclair Ltd.
25 Willis Road
Cambridge CB1 2AQ
England

Distributore
Rebit Computer
Via Induno, 18
20092 Cinisello B. (MI)
Tel. 02/6122371

Listino Prezzi:

Sinclair ZX Spectrum 16 Kbyte	*L. 360.000 + IVA*
Sinclair ZX Spectrum 48 Kbyte	*L. 495.000 + IVA*
Sinclair ZX Printer	*L. 195.000 + IVA*
Cassetta dimostrativa in italiano	*L. 48.000 + IVA*
Manuale in italiano	*L. 22.000 +* IVA

È inoltre disponibile una vasta scelta di cassette software in italiano.

Interfacciamento

Per la visualizzazione dei dati lo ZX Spectrum si collega direttamente ad un qualsiasi televisore, in bianco e nero o a colori, con l'apposito cavetto in dotazione. Chi lo volesse collegare direttamente ad un monitor può semplicemente collegarsi a monte del modulatore con una semplicissima modifica.
Chi è gia "amico" della famiglia Sinclair si ricorderà certamente il vecchio modo di memorizzare i programmi su nastro. Anche su questo punto siamo rimasti subito molto favorevolmente impressionati dagli enormi progressi fatti.
Il registratore si interfaccia al solito modo con i due cavettini in dotazione. Va bene qualunque registratore tenuto in ordine, stereofonico o monofonico, portatile o meno, a patto che abbia una presa per il microfono esterno e una presa per la cuffia tipo jack 3,5 mm. Come sullo ZX 80 bisogna staccare il jack EAR quando si registra; sarebbe stato preferibile un meccanismo automatico.
Su cassetta possono essere creati quattro diversi tipi di file, chiamati: *PROGRAM, BYTE, CHARACTER ARRAY* e *NUMBER ARRAY*. I file tipo *program* non sono nient'altro che l'unico tipo di file esistente sullo ZX 80 e 81, e contengono un programma e i valori delle relative variabili al momento della memorizzazione. I file tipo array sono la copia su cassetta di una certa matrice, ed essendo un modo molto comodo di memorizzare anche grosse quantità di dati indipendentemente dal programma, non potranno non essere ben accolti dagli "amici Sinclair". I file tipo *byte* sono esattamente ciò che indica il nome, cioè la copia fotografica di un certo numero di byte della memoria. Per quanto siano di uso un po' più sofisticato, sono senz'altro i più versatili e consentono, tra l'altro, di memorizzare una routine in Assembler o un'immagine video, grafica o meno.
Si noti il sistema di memorizzazione in due blocchi, per cui, prima del file vero e proprio ne vengono memorizzati in modo separato il nome, il tipo, la lunghezza ed eventualmente qualche altra informazione. Inoltre durante la ricerca e il caricamento dei file, a differenza dello ZX 80 e 81, l'immagine rimane stabile e, mentre sul bordo si alternano strisce di diversi colori che esternano le operazioni in esecuzione, nella parte centrale viene stampato il nome e il tiipo di ogni file incontrato. La ricerca di un file può iniziare da un qualunque punto del nastro prima di esso: fino a quando non viene identificata l'intestazione del file cercato, vengono ignorati tutti i "rumori" provenienti dal registratore.
I comandi utilizzabili con la cassetta sono: *SAVE* che serve ad archiviare un file; *LOAD* che carica un file già registrato; *VERIFY* da usare dopo *SAVE* per controllare la bontà della registrazione, cosa di gran lunga preferibile per non rischiare di perdere del lavoro prezioso; *MERGE* che esegue un *LOAD* senza però pulire prima la memoria, rendendo così possibile fondere due programmi. Sono molto interessanti le opzioni della *SAVE: LINE* in un comando *SAVE* fa sì che il programma memorizzato parta automaticamente, appena caricato, dalla line specificata (siccome viene eseguito un *GO TO* e non un *RUN* non vengono azzerate le variabili); *CODE* serve a specifiacare il byte di inizio e la lunghezza del segmento di memoria da copiare in un file tipo *BYTE; SCREEN$* è un abbreviazione per memorizzare l'immagine televisiva in un file tipo *BYTE; DATA* seguito dal nome di una matrice specifica un file tipo *ARRAY*.
E' stato annunciato ufficial-

mente un microdrive che oltre alle funzioni della cassetta userà: *PRINT, LIST, INPUT, INKEY$, OPEN$, CLOSE$, MOVE, ERASE, CAT* e *FORMAT*. Tutte queste funzioni sono già disponibili sulla tastiera. Abbiamo visto una fotografia del drive su un depliant inglese; è decisamente accattivante. Le prestazione dichiarate? 100 Kbyte per dischetto; velocità di trasferimento dati: 16 Kbyte al secondo (contro i già buoni 16 Kbyte in 100 secondi del registratore a cassette!); tempo di accesso medio 3,5 secondi, lento in assoluto ma ottimo in confronto alla cassetta.

Per un output permanente su carta, lo Spectrum usa la stessa ZX Printer dello ZX 81. Si tratta di una piccola (cm. 14x7,5x5) stampante termica su 32 colonne a 9 righe per pollice, che usa una carta argentata larga 10 cm. Si collega con un cavetto piuttosto corto, che termina con un connettore piatto a 2x22 poli. Unica facility un'indispensabile tastino di avanzanento posto sul lato destro. Stampando con parecchio rumore e una discreta qualità a 50 caratteri al secondo, impersona la stampante senza pretese per chi non vuole patire la volatilità del video. Se ne apprezzano il costo estremamente contenuto, corrispondente alle prestazioni, e la flessibilità dei comandi impiegabili. *COPY* esegue semplicemente l'hard copy del video, con grafici e tutto. *LLIST* è l'esatto equivalente di *LIST* e permette di stampare un listato di un programma BASIC. *LPRINT* prepara il buffer di stampa di una riga. Con quest'ultima, *TAB* ha lo stesso effetto che ha sul video, mentre *AT* sposta la posizione di stampa in un punto del buffer a piacere. E' comunque possibile collegare, tramite un apposita interfaccia, una stampante a carta normale.

Le ultime istruzioni per comunicare col mondo esterno sono la *IN* e la *OUT*, che controllano l'I/O a livello processore con la porta specificata. Idealmente ci possono essere fino a 65.536 porte diverse, alcune delle quali sono già usate per funzioni interne; otto leggono la tastiera e controllano contemporaneamente il registratore, una controlla la stampante, ecc. I primi cinque bit degli indirizzi di I/O sono usati per delle funzioni speciali; chi pensa di costruirsi un proprio dispositivo da interfacciare dovrà curare che il suo indirizzo abbia tutti questi bit a 1.

CARATTERISTICHE TECNICHE DELLO ZX SPECTRUM

- ROM 16 Kbyte usata per sistema operativo e BASIC.
- RAM 16 Kbyte (espandibili con 32 Kbyte) o 48 Kbyte, di cui: 6912 byte per gestione video (8x8 byte per ognuna delle 768 posizioni carattere + 768 byte per gli attributi); 256 byte per buffer printer; 182 byte per variabili del sistema.
- Video con 24 linee di 32 caratteri, le linee 22 e 23 usate per i comandi e le linee da 0 a 22 a disposizione del programma. Scrolling automatico e controllabile. Risoluzione grafica 256x192 pixel singolarmente indirizzabili.
- 8 colori disponibili per sfondo, bordo e scrittura, controllo luminosità, intensità e lampeggio.
- BEEP sonoro che può coprire più di 10 ottave.
- Tastiera con 40 tasti di gomma. Repeat automatico su tutti i tasti. Tutte le funzioni sono leggibili sulla tastiera.
- Set caratteri ASCII con minuscole e maiuscole, 20 caratteri grafici programmati e 21 programmabili.
- Collegamento con ZX Printer, 32 caratteri per linea, 9 linee per pollice, 50 caratteri al secondo.
- Annunciata interfaccia RS 232.
- Annunciati microdischi da 100 Kbyte con possibilità di collegamenti multipli.
- Collegamento diretto al registratore a nastro; trasferimento a 1500 baud. In 100 secondi trasferisce 16 Kbyte.
- BASIC esteso con possibilità di gestire file su cassetta e su floppy.

Hardware

Lo ZX Spectrum è basato sul diffusissimo microprocessore Z80A, ma la caratteristica saliente del circuito è un nuovo chip, costruito apposta per la Sinclair, denominato ULA, che scarica completamente lo Z80 dalle gestione del video, consentendo di avere un'immagine sempre assolutamente stabile, e contiene anche gran parte della logica necessaria allo stesso.

Aprendo la scatola si ha l'impressione di un hardware molto semplice e ordinato. Dopo aver rimosso le cinque viti sul fondo della scatola ci siamo trovati alle prese con i collegamenti della tastiera, due connessioni piatte flessibili che si sfilano dai relativi connettori con estrema facilità. Risaltano subito all'occhio il modulatore a colori, in alto a sinistra, e l'ingombrante dissipatore piatto del regolatore di tensione, in basso a destra, che sembra quasi proteggere il trasduttore incorporato. Con iniziale disappunto, abbiamo notato che lo Spectrum emette sempre un lievissimo rumore, come un cicalio, non fastidioso durante l'uso, comunque. Ci siamo ricreduti quando ci ha impedito di dimenticarci acceso il calcolatore un paio di volte. Tale ronzio è dovuto al survoltore che eleva la tensione a −12 V per alimentare la RAM dinamica.

Passando ad esaminare il circuito stampato, si notano tre grossi chip in posizione quasi centrale e una fila orizzontale di otto chip piccoli in basso a destra. I due piu grossi sono il microprocessore Z80A e l'ULA che si distingue grazie alla vistosa scritta Ferranti ULA, che ne indica il fabbricante. Quello largo un po' più corto è la ROM che contiene tutto il BASIC, il sistema operativo e il firmware necessario alla CPU (Z80A). Gli otto allineati sono le memorie RAM, per il software dell'utente. Lo Spectrum viene fornito con 16 o 48 Kbyte di RAM, comunque moltissimo per una macchina di dimensioni e costo dello Spectrum. La versione a 16 Kbyte può essere espansa a 48 Kbyte con una modifica interna. A parte resistenze e componenti vari, ci sono altri chip che costituiscono la logica sparsa, ridotta al minimo, necessaria al funzionamento del tutto.

Guardando il calcolatore dal lato posteriore si vedono tutte le connessioni predisposte; da si-

nistra a destra: la presa per l'alimentazione non stabilizzata a 9 Volt, il connettore piatto per stampante, dischi, I/O, ecc., gli spinotti tipo jack 3,5 mm. MIC e EAR, rispettivamente per la presa microfono e auricolare del registratore, la presa PLUG RCA del segnale video modulato per il collegamento diretto con un televisore. Si noti che, differentemente dai primi Sinclair, lo spinotto dell'alimentazione non è dello stesso tipo di quelli per il registratore e quindi non si rischia di fare collegamenti errati. Il connettore piatto (2x22) contiene praticamente tutti i segnali del bus Z80, oltre ad altri propri dello Spectrum che consentono di realizzare interfacciamenti di vario tipo.

Primissimo contatto

Aprendo la scatola si ha una sensazione di confidenza data l'assenza di cavi "difficili" o di gadget fantascientifici. Solo lui, il suo alimentatore, che potrebbe essere benissimo quello di un giocattolo o di una calcolatrice, e due cavetti, quasi da Hi-Fi: uno, doppio, per il registratore, l'altro, singolo, per la televisione.
L'installazione è veramente molto semplice, alla portata di tutti. Si inserisce l'alimentatore nella spina e se ne collega il filo all'apparecchio; si infila nello stesso il cavetto per la televisione e lo si attacca allo presa marcata UHF su quest'ultima; ci si sintonizza sul canale 36 UHF, come per cercare una stazione. L'unica difficoltà potrebbe sorgere da una incompatibilità tra la presa sulla televisione e il cavetto in dotazione, ma il rivenditore dovrebbe essere in grado di fornire un adattatore dopo una descrizione della presa stessa.
Terminate queste operazioni appare uno schermo bianco con in basso una scritta nera di copyright, che riappare dopo ogni *NEW*.

ERRATA CORRIGE

L'autore (L. Rizzo) dell'articolo "Contabilità casalinga con Sinclair (Bit n. 36) ci invia queste note correttive.

Le linee da modificare sono le 1550, 2045, 4130 e 7100 e vanno corrette come segue:

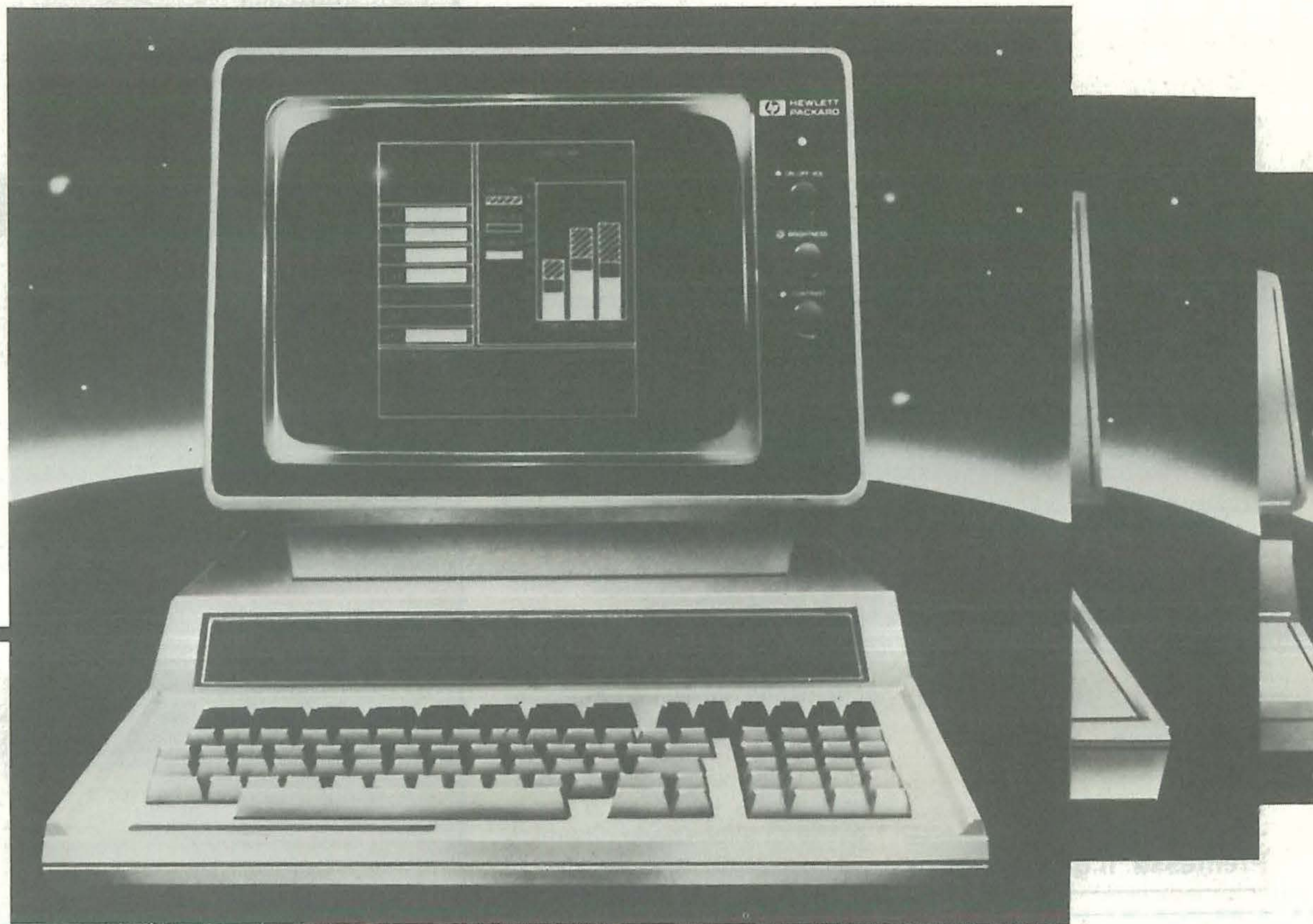
```
1550 PRINT TAB (I<10); I; TAB 3; S$
(I); "   =   "; TAB (31-LEN STR$ S(G,I
));S (G,I)

2045 LET I$ = "MESE   " +STR$ M

4130 PRINT AT 20,0; "      ART.  ?   (
  0   =   MENU   )"; X$ (1 TO 9); X$

7100 PRINT AT 19,0; "<"; S$ (N, TO
N(N); ">  :   NUOVO   NOME   ?"
```

Manuali

E' già disponibile un ottimo manuale in italiano edito dal Gruppo Editoriale Jackson, che è una versione in italiano, più che una traduzione, del manuale inglese della casa madre.
Articolato in due sezioni per un totale di 35 capitoli e 6 appendici, indica due approcci diversi: uno per chi non sa praticamente niente di calcolatori e annessi e vuole essere guidato tastotasto alla scoperta dell'informatica; l'altro per chi, già pratico di questi marchingegni, vuole avere spiegazioni complete, senza dover necessariamente procedere secondo un ordine prestabilito. Il libro si presta comunque bene ad una lettura sequenziale completa, con possibilità di saltare le parti giudicate non interessanti o troppo difficili. Vi sono anche parti dedicate a chi ignora completamente il pensiero aritmetico e matematico.
Degna di attenzione particolare è l'appendice D, che comprende 14 programmi-esempio completamente spiegati e provati, che sono stati fotografati direttamente dal listato della stampante. Consentono di verificare in pratica quanto si è appreso dalla lettura e di vedere, per chi non lo sa, come si può sviluppare un programma.
In ultimo, ogni capitolo è completato da esercizi che, più che essere orientati a risolvere uno specifico problema, tendono a mettere in risalto i particolari più importanti che una volta acquisiti impediranno di ripetere errori, a volte banali, in successive e personali stesure di programmi.

Conclusioni

Lo ZX Spectrum ha delle caratteristiche di spiccata originalità e, secondo una tendenza sempre più diffusa, presenta un hardware ridotto al minimo indispensabile, che supporta un software di alto livello. In tal modo mette a disposizione di chiunque, semplicemente ed economicamente un orizzonte informatico più che ampio per una totale comprensione della materia, così consentendo, una risoluzione "programmatica" avanzata anche di problemi complessi, che potranno essere risolti a molti livelli con totale soddisfazione dell'acquirente. Il calcolatore, oggetto domestico di consumo, diventa sofisticato, cominciando a mostrare serio impegno e buona volontà per essere sempre più considerato strumento e non giocattolo.
Dal punto di vista strettamente tecnico merita un elogio incondizionato, giudizio che immediatamente stimola un confronto con i suoi pari. Le inevitabili limitazioni che lo fanno dichiarare "non idoneo fisicamente" al duro lavoro professionale, per il quale si è abituati a sentir parlare di molti milioni di costo; la più grave è l'assenza di una tastiera standard (e chi dice che non se ne possa interfacciare una esterna?), seguita da un video di sole 32 colonne e dalla mancanza di una memoria di massa molto capace, che costerebbe molte volte il prezzo della macchina. È d'obbligo, comunque, una riflessione sui progressi della tecnica dall'inizio dell'era elettronica, per poter intravedere i prossimi chilometri di un cammino che appare sempre più affascinante e di cui lo Spectrum ne è certamente segno encomiabile.
Dal punto di vista BASIC, la lingua avvelenata del perfezionista fa notare che non ci sarebbe voluto molto per poter avere anche i nomi delle variabili-stringa lunghi a piacere, o il comando *ELSE*, o delle vere matrici di stringhe,o un comando per colorare l'interno di figure chiuse senza ricorrere a strane programmazioni,o la possibilità di usare più colori in una stessa posizione carattere, o la possibilità di variabili numeriche intere o in doppia precisione, ecc. ecc. Forse, però, avere tutto questo sarebbe costato qualcosa o forse molto di più. La Sinclair, per ora, ha scelto la strada di un favorevole rapporto prezzo/prestazioni. L'aveva intrapresa quasi tre anni fa, e ripensiamo al Dicembre 1980, quando su queste stesse pagine (**Bit** n.13) un calcolatore di soli 200 grammi, chiamato ZX 80, ci entusiasmò pur potendo svolgere calcoli con i soli numeri interi! A ripensarci ora sentiamo in noi una grande tenerezza. ■

HP 86. Con tutte queste soluzioni non c'è piú spazio per i problemi.

Il nuovo HP 86: un *personal computer* che ti offre una gamma di soluzioni in grado di espandersi con le tue esigenze.
A un prezzo accessibile.

Soluzioni grafiche.

Dai diagrammi di vendita agli organigrammi, crei tutto ciò che ti serve per esporre nel modo più efficace ciò che devi dire. E tutto in pochi minuti.

Grafici circolari, istogrammi, lineari, diapositive di testi e trasparenti per lavagne luminose. A colori!

Soluzioni di calcolo e analisi.

Pensa al tempo che passi a compilare fogli di lavoro ripetitivi.

Con il *Software Visicalc® Plus,* basta impostare una variabile: al resto pensa l'HP 86. Istantaneamente.

Anche per le previsioni a lungo termine, che prima richiedevano l'uso di un grosso e costoso *computer.*

E poi, puoi trasformare i risultati in un grafico.

Soluzioni per lettere, memo, rapporti.

Potrai finalmente impostare rapporti senza bisogno di carta, finchè non trovi la forma migliore: scrivi e correggi rapidamente sul video. Personalizzi lettere per la *mailing-list,* e infine puoi usare una stampante HP per fare di ogni copia un "originale".

Soluzioni per la gestione delle informazioni.

Con l'HP 86 puoi creare il tuo archivio clienti, dalla *mailing-list* alle schede personali o alle registrazioni contabili.

Una ricerca ed un aggiornamento di dati come questi erano normalmente possibili solo con i grandi *computer.*

E soluzioni hardware.

La possibilità di espansione di un *computer* dipende in gran parte dal *software,* naturalmente *l'hardware* deve essere all'altezza della situazione.

Ecco perchè l'HP 86 è progettato modularmente. Tu compri soltanto le parti che ti interessano, e quando ne avrai bisogno potrai comodamente aggiungere le periferiche HP, certo di avere un omogeneo sistema integrato.

Un sistema che in più gode della eccellente rete di assistenza HP.

Quindi, se hai problemi, telefona alla Hewlett-Packard Italiana (02-90369468) e chiedi l'indirizzo del rivenditore HP più vicino; ti mostrerà la serie 80 e l'HP 86.

Quando sono i risultati che contano

HEWLETT PACKARD

> È il penultimo di tre articoli nei quali si discute in modo abbastanza approfondito l'implementazione di un'interfaccia per il controllo di unità a disco da 5"1/4. Dopo la prima parte introduttiva, essenzialmente didattica e la seconda dedicata all'hardware della scheda controller dei moderni chip LSI relativi, proseguiamo con la descrizione del software che presiede alla gestione delle varie transazioni sui dischetti, sotto forma di listato commentato (e **Bit** *ritiene doveroso ringraziare pubblicamente chi ha saputo calare da un pulpito professionale una così cristallina lezione divulgativa: è un incoraggiamento a proseguire ed un invito-rivolto anche ad altri professionisti - a fare altrettanto* NdR).

di P. Bozzola ing.
Computerjob Elettronica
Brescia

Parte terza

Premessa: il giusto mezzo

Giunti verso la dirittura d'arrivo, ovvero al termine di questa fatica un poco improba, sorge il dubbio che tra i lettori possa esserci qualche smaliziato che abbia per un verso trovato un po' troppo complessa (e magari ridondante) l'elettronica di controllo descritta nella scorsa puntata, per ritenere invece alquanto ridondante è magari esagerato il costrutto della RWTS che stiamo per esporre qui di seguito. Ebbene, quasi sicuramente si tratta di gente che appartiene alla categoria dei buoni conoscitori di DOS assai più evoluti e/o molto noti (il DOS dell'Apple, per dirne uno fra tanti), per cui un "K" dedicato alla RWTS sembrerà loro sprecato. Francamente, sotto il puro profilo tecnico non si può negare che una simile tesi non fa una grinza. Ma qui è giocoforza tornare a ribadire che il progetto e quanto è qui descritto ha soprattutto finalità didattiche. Forse anche quei tali venticinque (o 250?) lettori esperti converranno che, per molti altri sarebbe stato un colpo basso sottoporre loro sofisticatezze come: interleaving, gestione via software del segnale letto-/scritto da/su disco, ottimizzazione della ricerca del settore, e altre analoghe, tipiche di DOS che usano hardware e software più evoluti e, proprio per questo di ben più ardua comprensibilità. A voler poi parlare di tecniche anche interessanti come quelle di allocazione dinamica ed ottimizzata dei file della crittografia (sui dati e sulle stesse stringhe di sincronizzazione!) e di altre ancora sarebbe stato necessario un vero e proprio trattato in più volumi. Riservandomi, magari, di tornare su tali argomenti in successivi articoli, per il momento ritengo (o, perlomeno, spero) che la semplicità del progetto, l'abbondanza di ripetizioni (anche nel software), e persino il modo qua e là un tantino semplicistico con cui si è scelto di trattare l'argomento, abbiano permesso ai lettori meno addentro a tali segrete cose d'iniziare in modo "soft" la penetrazione di certi misteri relativi all'utilizzo delle unità a dischi. (*Così questi appariranno a tutti meno arcani e il know how conseguito servirà a molti come trampolino di lancio verso più ambiziosi traguardi: ben detto!* NdR).

Concetti base sul DOS

La prima cosa, necessaria per lavorare con una unità a disco, è (oltre naturalmente all'interfaccia) un "blocco" di software

a)

TIPO	COMANDO	7	6	5	4	3	2	1	0
I	Restore	0	0	0	0	h	V	r_1	r_0
I	Seek	0	0	0	1	h	V	r_1	r_0
I	Step	0	0	1	u	h	V	r_1	r_0
I	Step In	0	1	0	u	h	V	r_1	r_0
I	Step Out	0	1	1	u	h	V	r_1	r_0
II	Read Command	1	0	0	m	b	E	0	0
II	Write Command	1	0	1	m	b	E	a_1	a_0
III	Read Address	1	1	0	0	0	E	0	0
III	Read Track	1	1	1	0	0	1	0	s
III	Write Track	1	1	1	1	0	1	0	0
IV	Force Interrupt	1	1	0	1	I_3	I_2	I_1	I_4

b) TIPO 1

h = Head Load flag (Bit 3)

h = 1, Load head all'inizio
h = 0, Load head non all'inizio

V = Verify flag (Bit 2)

V = 1, Verifica all'ultima traccia
V = 0, Nessuna verifica

r_1 r_0 = Stepping motor rate (Bit 1-0)

Cfr. Tav. 1 per il sommario delle velocità

u = Update flag (Bit 4)

u = 1, Update Track register
u = 0, Nessun update (aggiorn.to)

c) TIPO 2

m = Multiple Record flag (Bit 4)

m = 0, Singolo record
m = 1, Record multiplo

b = Block length flag (Bit 3)

b = 1, formato IBM (da 128 a 1024 byte)
b = 0, formato non IBM (da 16 a 4096 byte)

a_1 a_0 = Data Address Mark (Bit 1-0)

a_1 a_0 = 00, FB (Data Mark)
a_1 a_0 = 01, FA (User defined)
a_1 a_0 = 10, F9 (User defined)
a_1 a_0 = 11, F8 (Deleted Data Mark)

d) TIPO 3

s = Sincronizzazione flag (Bit 0)

$\overline{s}$ = 0, Sincronizzazione ad AM
$\overline{s}$ = 1, Non sincronizzazione ad AM

TIPO 4

Ii = Interrupt Condition flags (Bit 3-0)
I_0 = 1, Not Ready to Ready Transistor
I_1 = 1, Ready to Not Ready Transistor
I_2 = 1, Index Pulse
I_3 = 1, Immediate interrupt

E = Enable HLD e ritardo di 10 msec
E = 1 Enable HLD HLT e ritardo di 10 msec
E = 0 La testina è supportata e non si ha ritardo di 10 msec.

NOTA: L'FDC ha le linee di DATABUS che INVERTONO (sia in ingresso che in uscita), per cui i dati ed i comandi vanno COMPLEMENTATI. (I bit di queste tavole sono anch'essi da complementare).

Tavola 1 - Elenco dei comandi con i rispettivi flag ed i loro significati.

I floppy disk controller: teoria e pratica di progetto hard/soft

comunemente chiamato Read/Write Track/Sector in sigla RWTS. Si tratta in sostanza di una routine che compie nè più nè meno quanto è descritto nella parte introduttiva del listato che correda il presente articolo. Occorre innanzitutto precisare che, ovviamente, è assai scomodo per l'utente dover dire alla RWTS, tutte le volte, il numero del driver, il settore e la traccia da leggere/scrivere, oltre al fatto che, appunto si voglia leggere o scrivere, e così via. Altrettanto scomodo risulta il dover ad esempio verificare a posteriori i risultati andando a leggere in RAM la cella dell'eventuale codice di errore. Nasce così la necessità di un *gestore* costruito attorno al nucleo costituito dalla RWTS in modo tale da consentire un facile ed immediato dialogo con l'utente (in genere in modo interattivo). È quanto oggi si tende a inquadrare nella categoria delle interfacce d'utente, ma che, nella fattispecie è in pratica il "*Disk Operating System*". Di esso diamo ora alcuni ragguagli, focalizzando le caratteristiche operative più importanti, ossia quello che *veramente* serve.

In primo luogo un DOS deve essere strutturato rigorosamente, possedendo in particolare un unico punto di uscita. In tal modo potrà essere usato come se fosse una subroutine, (se ne accennò anche nella prima puntata) e l'utente potrà accedervi in modo "pulito" da un ambiente operativo qualsiasi: da BASIC, da Assembler, da Editor, da Pascal, da FORTH, ecc. Sarà infatti sufficiente che ognuno di tali environment possieda almeno un punto di "link" per l'aggiunta di comandi esterni, semprechè addirittura non

DIMENSIONE	DENSITA'	NUMERO FACCE	CAPACITA' (NOMINALE) NON-FORMATTATO		TEMPO DI TRASFERIMENTO PER BYTE	CAPACITA' FORMATTATO	
			PER TRACCIA	PER DISCO		PER TRACCIA	PER DISCO
5 1/4"	SINGOLA	1	3125	109,375*	64μs	2304**	80,640
5 1/4"	DOPPIA	1	6250	218,750	32μs	4608***	161,280
5 1/4"	SINGOLA	2	3125	218,750	64μs	2304	161,280
5 1/4"	DOPPIA	2	6250	437,500	32μs	4608	322,560
8"	SINGOLA	1	5208	401,016	32μs	3328	256,256
8"	DOPPIA	1	10,416	802,032	16μs	6656	512,512
8"	SINGOLA	2	5208	802,032	32μs	3328	512,512
8"	DOPPIA	2	10,416	1,604,064	16μs	6656	1,025,024

* Basato su 35 traccia/lato
** Basato su 18 settori/traccia (128 byte/sec.)
*** Basato su 18 settori/traccia (256 byte/sec.)

Tavola 2 - Capacità di memorizzazione su floppy disk per 5" 1/4 e per 8", in singola e doppia densità.

Figura 1 - Un esempio di applicazione dei nuovi chip di interfaccia.

PROFESSIONAL 325

RAINBOW 100

PROFESSIONAL 350

Un Personal Computer Digital al prezzo di un personal computer. Dove trovarlo?

Ecco l'elenco dei primi rivenditori dove potrete trovare i famosi Personal Computers Digital: una combinazione unica di qualità, prestazioni e servizio:

PUNTI DI VENDITA IN LOMBARDIA. Rivenditori Digital: AG Informatica 02/4983416 - Basic Computers 02/8242970 - Happy System 02/802433 - Homic Personal Computer 02/4988201 - Softec 02/7491196 - Control System 0372/27012
Rivenditori Celdis (Distributore Autorizzato Nazionale): Adco Informatica 02/719267 - Computer Solution 02/2139957 - Indis 02/8243451 - Siro 02/6070437 - Symic 02/2130450 - Antec Computer 0376/398759 - EMI 039/386152 - Informatica EDP 0342/213227 - Italian General Consultants 0331/514134 - Iter 035/216313 - I.R.P.E. 0332/238533 - M.A.S.H. Computer - System 0382/37300 - P.W.S. Informatica 030/221133 - Tecnomation 031/703869

PUNTI DI VENDITA IN PIEMONTE. Rivenditori Digital: Softec 011/8396444
Rivenditori Celdis (Distributore Autorizzato Nazionale): Aba Elettronica 011/332065 - Mesar Team 011/761382 - Software Engineering 011/503746 - Tesi 011/658536 - Kernel 0121/72455

PUNTI DI VENDITA IN VALLE D'AOSTA. Rivenditori Celdis (Distributore Autorizzato Nazionale): Informatique 0165/2242

PUNTI DI VENDITA IN LIGURIA. Rivenditori Digital: T.P. 010/280276
Rivenditori Celdis (Distributore Autorizzato Nazionale): Compufficio 010/564493 - MMC 010/886422 - Computer Centre 0185/314142 - Office Automation 0185/304747 - Computer House 0183/650774 - Riviera Computer 0184/86929

PUNTI DI VENDITA IN VENETO. Rivenditori Celdis (Distributore Autorizzato Nazionale): GP Dati 041/987122 - H.S.H. 049/663888 - Sic Italia 049/22820 - Mos 80 045/914400 - S.I.C. CO. 0444/502115

PUNTI DI VENDITA IN FRIULI. Rivenditori Digital: Data Shop 0434/27338
Rivenditori Celdis (Distributore Autorizzato Nazionale): Arco Inf. 040/62440 - Computer Centre 040/574090 - Di Giusto Marcuzzi 0432/482547

PUNTI DI VENDITA IN TRENTINO. Rivenditori Celdis (Distributore Autorizzato Nazionale): Società Nazionale Servizi 0461/650434

PUNTI DI VENDITA IN EMILIA ROMAGNA. Rivenditori Digital: Canalgrande Informatica 059/219801
Rivenditori Celdis (Distributore Autorizzato Nazionale): Recom 051/534883 - EDP System 059/554487 - Spazio Dati 059/800654 - D.S. Data System 0521/206084 - Multistudio 0523/37839

PUNTI DI VENDITA IN TOSCANA. Rivenditori Digital: F.lli Bassilichi 055/439841 - Euro Computer 0585/488663
Rivenditori Celdis (Distributore Autorizzato Nazionale): ZEA Computer 055/295476 - Data Port 0587/53858 - IT-Lab 050/501359 - Logos Informatica 0583/584141

PUNTI DI VENDITA IN UMBRIA. Rivenditori Digital: F.lli Bassilichi 075/751849

PUNTI DI VENDITA IN LAZIO. Rivenditori Digital: Bit Computers 06/ 5126700 - Codat 06/634841 - Computer Center 06/875638 - GEA 06/872608 - Genel 06/6235464 - Codat 0746/44704
Rivenditori Celdis (Distributore Autorizzato Nazionale): Assocomputers 06/3561123 - Eurocom 06/7574487 - Hard Soft 06/8277500 - Memo 06/3607457

PUNTI DI VENDITA IN ABRUZZO. Rivenditori Digital: Codat 085/932411 - Genel 085/376142 - GEA 0861/54712 - GEA 0862/71121
Rivenditori Celdis (Distributore Autorizzato Nazionale): CPS Informatica 0862/62558 - Memo 0863/23845

PUNTI DI VENDITA IN CAMPANIA. Rivenditori Digital: Codat 081/241866 - Golden Computers 081/378634 - Texime 081/629205

PUNTI DI VENDITA IN PUGLIA. Rivenditori Digital: Ciesse 080/365938 - GEA 080/451098 - Codat 0832/49093

PUNTI DI VENDITA IN CALABRIA. Rivenditori Digital: Sirangelo Computers 0984/75741
Rivenditori Celdis (Distributore Autorizzato Nazionale): Gestione Elettroniche Aziendali 0965/22973 - Tecnocomp 0966/51818

PUNTI DI VENDITA IN SICILIA. Rivenditori Digital: SI.PR.EL. 091/577344
Rivenditori Celdis (Distributore Autorizzato Nazionale): Siro 091/250579 - Computer Shop 095/441620 - SI.EL. 095/322875 - Hardware Software Service 090/775912

PUNTI DI VENDITA IN SARDEGNA. Rivenditori Digital: Passamonti S.p.A. 070/662541
Rivenditori Celdis (Distributore Autorizzato Nazionale): Computer Systems & Electronics 070/485996

DISTRIBUTORE AUTORIZZATO NAZIONALE PER RIVENDITORI:
CELDIS ITALIANA S.p.A. Via F.lli Gracchi, 36 - 20092 CINISELLO BALSAMO (MILANO) Tel. 02/6120041

DIGITAL EQUIPMENT S.p.A. V.le Fulvio Testi, 11 - 20092 CINISELLO BALSAMO (MILANO) Tel. 02/617961, TORINO 011/7413001, ROMA 06/546881, PADOVA 049/633877

ne siano già forniti, cosa che peraltro avviene molto spesso nei linguaggi, fin dall'inizio.
A parte ciò però il problema fondamentale è *la memorizzazione dei file (dati e/o programmi) sul disco, con criterio della loro distribuzione ordinata e facilmente accessibile per una rapida rilettura (recovery).* Tale compito non è affatto facile. Innanzitutto infatti, occorre riservare (ovviamente sul disco stesso) un certo spazio per l'"indice", ossia, un vero e proprio sommario che indichi chiaramente il contenuto del disco. L'indice deve essere autonomamente gestito dal DOS, per quanto riguarda gli aggiornamenti necessari ogni volta che si aggiunge un nuovo file oppure se ne elimina uno preesistente. Inoltre all'utente dovrà essere offerta la possibilità di visionare l'indice (o "catalogo" o "directory" come è sovente chiamato) tutte le volte che lo desidera.
Facciamo allora un esempio di organizzazione del dischetto. Se esso ha 40 tracce, potrà esser conveniente riservare all'indice la traccia 20, perchè così vengono diminuiti, in media, gli spostamenti della testina per la ricerca (seek) dei settori. Su questa traccia mediana si potranno organizzare i settori in modo che contengano, distribuiti secondo un criterio opportuno (col fine, cioè che la ricerca "table search type" risulti rapida ed efficiente), i nomi dei file memorizzati sul disco. Per il momento - in accordo con quanto detto nella premessa - supponiamo per semplicità di lavorare con un sol tipo di file. In ogni caso accanto al nome del file, dovrà esserci un indirizzo, espresso nella forma "traccia + settore", che punti ad un determinato settore del disco stesso. In quel settore non troveremo ancora il file, bensì una "lista", ovvero un dettagliato elenco di tutti i settori, sparsi sul disco, che compongono, finalmente, quel file. In conclusione la ricerca di un ipotetico file denominato *ALFA1* seguirà le fasi qui di seguito elencate:

1) ricerca nell'indice del nome "*ALFA1*";
2) trovatolo, subito accanto verrà letto il puntatore al (primo) settore della "lista";
3) ci si porta sul settore individuato nella fase precedente e lo si legge;
4) disponendo della lista, ora si può procedere alla ricomposizione del file, il che si effettua "pescando" i vari settori ad uno ad uno e scrivendoli in memoria l'uno di seguito all'altro.

Dove sarà posto, nella memoria di sistema, il file così ricomposto? Ovviamente a partire da un dato indirizzo e per una determinata lunghezza, parametri che, ad esempio, potrebbero trovarsi in testa alla predetta "lista". Occorre qui precisare che non ci si soffermerà nè sui criteri di distribuzione - sul disco - dei settori, nè sui metodi ottimizzati di ricerca e di occupazione (vale a dire: dei settori liberi sparsi qua e là, quali sono da riempire per primi mentre si sta salvando un file?) nè su altre faccende del genere. Dovrebbe tuttavia risultare perlomeno intuitivo il fatto che, una volta accettata la RWTS come *nucleo* del nostro DOS, un grosso lavoro sarà fatto da questa routine, cui una routine più esterna, ad esempio una *SEARCH* per la lettura o una *SAVE* per la scrittura, passano - settore per settore - i parametri dovuti. Le ultime due possono addirittura essere pensate come due blocchi "evoluti" che necessitino di conoscere solo il nome del file la prima e anche gli indirizzi iniziale e finale in RAM la seconda. Tali parametri verranno introdotti dall'utente dialogando col DOS, quando quest'ultimo si trova nello stato "comandi".
La frammentazione dei file è stata fin dall'inizio data per scontata: essi sono cioè, soggetti ad una diaspora continua, al limite del più sadico dei maltrattamenti. Per quanto possa sembrare caotico, tutto ciò, risulterà invece procedimento molto assennato non appena si esamini che cosa accade fin dal momento del salvataggio sul disco del primo file. Immaginiamo un'ipotetica storia del disco che ha inizio con la traccia indice ancora vuota (condizione "No Index") ed il disco completamente libero. Ciò sarà estremamente visibile grazie ad un apposito spazio - dell'indice stesso - dedicato alla "Vo-

```
                0020            ;
                0030            ; ****************************
                0040            ;   R   W   T   S
                0050            ; ****************************
                0060            ;
                0070            ; 65XX-DISK READ/WRITE TRACK-SECTOR ROUTINE
                0080            ;
                0090            ;
                0100            ; QUESTA SUBROUTINE GESTISCE COMPLETAMENTE
                0110            ; UNA INTERFACCIA PER DISCHETTI DA 5"1/4, SE
                0120            ; SI USA LA SCHEDA "MINIMICRO FDC".
                0130            ; LA SUBROUTINE VA CHIAMATA CON I SEGUENTI
                0140            ; PARAMETRI IN INGRESSO:
                0150            ;        1) R/WFLAG = 1 PER LETTURA
                0160            ;           SUL DISCHETTO, 0 PER SCRITTURA;
                0170            ;        2) NDISK = 1 o 2 (NUMERO DRIVER)
                0180            ;        3) NSCHEDA = 0 (TIPICAMENTE)
                0190            ;        4) NTRACK = NUMERO TRACCIA OVE SI
                0200            ;           INIZIA LA TRANSAZIONE;
                0210            ;        5) NSECTOR = NUMERO SETTORE INI=
                0220            ;           ZIALE SU QUELLA TRACCIA
                0230            ;        6) INITADDR (2 BYTES) = INDIRIZZO
                0240            ;           INIZIALE DELLA ZONA DI RAM INTE=
                0250            ;           RESSATA NELLA TRANSAZIONE;
                0260            ;        7) ENDADDR (2 BYTES) = INDIRIZZO
                0270            ;           FINALE DELLA STESSA ZONA DI RAM
                0280            ;
                0290            ; NEL CASO LA TRANSAZIONE SIA COMPIUTA
                0300            ; SENZA ERRORI, LA LOCAZ. "ERRCODE" CONTERRA'
                0310            ; "$FF", MENTRE NTRACK ED NSECTOR SARANNO STATI
                0320            ; AGGIORNATI "AL PRIMO SETT. LIBERO E RELATIVA
                0330            ; TRACCIA, SUL DISCO, DOPO IL FILE AP=
                0340            ; PENA TRATTATO".
                0350            ; NEL CASO DI ERRORE, LA TRANSAZIONE E'
                0360            ; ABORTITA, VIENE RILEVATO L' ERRORE IL
                0370            ; CUI CODICE (SEMPRE ( $80) VA IN "ERRCO=
                0380            ; DE". NTRACK ED NSECTOR SONO AGGIORNATI
                0390            ; ALL' ULTIMO PUNTO FINO AL QUALE NON
                0400            ; C' E' STATO ERRORE, ED IN OGNI CASO SI
                0410            ; SPENGONO I MOTORI.
                0420            ; LE VARIE SUBROUTINE USATE SONO CITATE
                0430            ; ANCHE COL LORO INDIRIZZO CON CUI COM=
                0440            ; PAIONO NEL DOS PER IL SYM. NEL CASO IN
                0450            ; CUI L' INDIRIZZO SIA SEGUITO
                0460            ; DA "*****" CIO' INDICA CHE
                0470            ; QUELLA ROUTINE HA SUBITO MO=
                0480            ; DIFICHE RISPETTO ALL' ORIGI=
                0490            ; NALE DEL D.O.S. PER IL SYM.
                0500            ;
                0510            ; COPYRIGHT (C) COMPUTERJOB EL. , 1982
                0520            ;
                0530            ; ****************************************
                0540            ;
                0550            ;
                0560            ; CELLE DI PAGINA ZERO (WORK PARMS)
                0570            ;
                0580            .BA $0000
                0590            ;
0000-           0600 R/WFLAG    .DS 1
0001-           0610 NDISK      .DS 1
0002-           0620 SAVEPARM   .DS 1
0003-           0630 NSCHEDA    .DS 1
0004-           0640 FDCCOM     .DS 1
0005-           0650 ERRCODE    .DS 1
0006-           0660 NDRIVER    .DS 1
0007-           0670 FDCSTATUS  .DS 1
0008-           0680 NTRACK     .DS 1
0009-           0690 NSECTOR    .DS 1
000A-           0700 DRSTATUS   .DS 1
000B-           0710 INITADDR   .DS 2
000D-           0720 ENDADDR    .DS 2
000F-           0730 NTENT      .DS 1
0010-           0740 NERR       .DS 1
                0750 CONF       .DE DRSTATUS
0011-           0760 BASE       .DS 2
                0770            ;
                0780            .BA $C000
                0790            .OS
                0800            ;
C000- 4C F9 C3  0810 START      JMP START1    ; RINVIO AL PROGR. VERO
                0820            ;
                0830 VIA        .DE $A880     ; 6522 BASE ADDRESS
                0840 DRB        .DE VIA
                0850 DRA        .DE VIA+1
                0860 DDRB       .DE VIA+2
                0870 DDRA       .DE VIA+3
                0880 T1CL       .DE VIA+4
                0890 T1CH       .DE VIA+5
                0900 T1LL       .DE VIA+6
                0910 T1LH       .DE VIA+7
                0920 T2CL       .DE VIA+8
                0930 T2CH       .DE VIA+9
                0940 ACR        .DE VIA+$B
                0950 PCR        .DE VIA+$C
                0960 IFR        .DE VIA+$D
                0970 IER        .DE VIA+$E
                0980            ;
                0990            ; VALORI PER CTRLBLK
                1000            ;
                1010 INIT       .DE $10       ; INIZIALIZZA, DEFAULT DRIVER 2
                1020 INIT1      .DE $30       ; IDEM, MA DRIVER 1
                1030 DEINIT     .DE $20       ; SPEGNI TUTTO !
                1040 READCOMP   .DE $00       ; READ & COMPARE
                1050 WRITCOMP   .DE $01       ; WRITE & COMPARE
                1060 READ       .DE $03       ; READ
                1070 WRITE      .DE $04       ; WRITE
                1080 COMPARE    .DE $06       ; COMPARE
                1090            ;
                1100            ; **********************
                1110            ;      SUBROUTINES
                1120            ; **********************
                1130            ;
                1140            ; SUB 7631******
                1150            ; PREPARA NELLA LOC. NDRIVER
                1160            ; IL VALORE CHE POI SARA'
                1170            ; SCRITTO NEL PORT B PER A=
                1180            ; ZIONARE DRIV.1,2 O SPEGNERLI.
                1190            ;
C003- A9 00     1200 SETDRIVE1  LDA #0
C005- 2C        1210            .BY $2C       ; TRUCCO:FALSO "BIT"
C006- A9 02     1220 SETDRIVE2  LDA #02
C008- 2C        1230            .BY $2C       ; FALSO "BIT"
C009- A9 06     1240 DRIVEOFF   LDA #06
C00B- 85 06     1250            STA *NDRIVER
C00D- 60        1260            RTS
                1270            ;
                1280            ; SUB 7BAC
                1290            ; "VIAINIT" INIZIALIZZA LA
                1300            ; VIA DELLA FDC ED ESEGUE
                1310            ; UN "FORCE INTERRUPT" AL
                1320            ; FLOPPY DISK CONTROLLER.
                1330            ;
C00E- 20 C1 C0  1340 VIAINIT    JSR IRQDIS    ; DISABILITA IRQ DA VIA
C011- A9 E0     1350            LDA #$E0
C013- 9D 8B A8  1360            STA ACR,X     ; PB7 FREERUN T1 E T2 CONTA DA PB6
C016- A9 FB     1370            LDA #$FB
C018- 9D 8C A8  1380            STA PCR,X     ; CB2=1, INTRQ,TRQ SET, CA2 PULSE
```

Seguito listato.

```
C01B- A9 C2     1390            LDA #$C2
C01D- 9D 86 A8  1400            STA T1LL,X    ; PREPARA TIMER1, T=390uS
C020- A9 00     1410            LDA #0
C022- 9D 85 A8  1420            STA T1CH,X
C025- 9D 88 A8  1430            STA T2CL,X    ; AZZERA T2
C028- 9D 89 A8  1440            STA T2CH,X
C02B- A9 00     1450 FORCEINT   LDA #0
C02D- 9D 83 A8  1460            STA DDRA,X    ; PA TUTTI INPUT
C030- A9 BF     1470            LDA #$BF
C032- 9D 82 A8  1480            STA DDRB,X    ; PB TUTTI OUT TRANNE PB6
C035- A9 26     1490            LDA #$26
C037- 9D 80 A8  1500            STA DRB,X     ; READ=1, DRA0,1=11 (MOTOR OFF)
C03A- BD 81 A8  1510            LDA DRA,X     ; CA2 PULSE & RESET FLIPFLOP
C03D- BD 8D A8  1520            LDA IFR,X     ; TRUCCO PER RESETTARE IFR
C040- 9D 8D A8  1530            STA IFR,X
C043- A9 2F     1540            LDA #$2F      ; "FORCE INTERRUPT" COMMAND
C045- 85 04     1550            STA *FDCCOM
C047- 20 A0 C0  1560            JSR EXECCOM   ; ESEGUE COMANDO
C04A- 60        1570            RTS
                1580            ;
                1590            ; SUB 7BEA
                1600            ; "PUNTSCHED" RICAVA IL
                1610            ; VALORE CORRETTO DI "X" A
                1620            ; PARTIRE DA NSCHEDA, IN
                1630            ; MODO CHE SI POSSANO USARE
                1640            ; PIU' SCHEDE FDC "IN PARAL=
                1650            ; LELO" MA SINGOLARMENTE
                1660            ; SORVEGLIATE. SI NOTI CHE
                1670            ; TUTTE LE OPERAZIONI SULLA VIA
                1680            ; SONO INDICIZZATE "X".
                1690            ;
C04B- A5 03     1700 PUNTSCHED  LDA *NSCHEDA          ; RICAVA "X" DA NSCHEDA
C04D- 0A        1710            ASL A
C04E- 0A        1720            ASL A
C04F- 0A        1730            ASL A
C050- 0A        1740            ASL A
C051- AA        1750            TAX
C052- 60        1760            RTS
                1770            ;
                1780            ; SUB 7BF3
                1790            ; "FORCE" FA ESEGUIRE UN IM=
                1800            ; PULSO A CB2 PER FORZARE
                1810            ; UNA LETTURA INDIPENDENTE=
                1820            ; MENTE DALL' HANDSHAKE OUT
                1830            ; DELLA VIA.
                1840            ;
C053- BD 8C A8  1850 FORCE      LDA PCR,X     ; CA2 PULSE PER FORZARE LETTURA
C056- 29 DF     1860            AND #$DF
C058- 9D 8C A8  1870            STA PCR,X
C05B- 09 20     1880            ORA #$20
C05D- 9D 8C A8  1890            STA PCR,X
C060- 60        1900            RTS
                1910            ;
                1920            ; SUB 7C01
                1930            ; "MOTOR" AGISCE SU PB
                1940            ; COL CONTENUTO DI NDRIVER,
                1950            ; SETTANDO COSI' PB1 & PB2
                1960            ; OPPORTUNAMENTE.
                1970            ;
C061- BD 80 A8  1980 MOTOR      LDA DRB,X
C064- 29 F8     1990            AND #$F8
C066- 05 06     2000            ORA *NDRIVER          ; ATTIVA DRIVER
C068- 4C 9C C0  2010            JMP EXESTA    ; (RTS)
                2020            ;
                2030            ; SUB 7C0C
                2040            ; "SETWRITE" PREPARA PA PER
                2050            ; UNA SCRITTURA AL FDC 1771
                2060            ;
C06B- A9 FF     2070 SETWRITE   LDA #$FF      ; PREPARA PER SCRITTURA IN FDC
C06D- 9D 83 A8  2080            STA DDRA,X
C070- BD 80 A8  2090            LDA DRB,X
C073- 29 DF     2100            AND #$DF      ; LINEA READ AL FDC VA A ZERO
C075- 4C 9C C0  2110            JMP EXESTA    ; (RTS)
                2120            ;
                2130            ; SUB 7C19
                2140            ; "SETREAD" PREPARA PA PER
                2150            ; UNA LETTURA DAL FDC 1771
                2160            ;
C078- A9 00     2170 SETREAD    LDA #0        ; PREPARA PER LETTURA DAL FDC
C07A- 9D 83 A8  2180            STA DDRA,X
C07D- BD 80 A8  2190            LDA DRB,X
C080- 09 20     2200            ORA #$20      ; LINEA READ AL FDC VA A UNO
C082- 4C 9C C0  2210            JMP EXESTA    ; (RTS)
                2220            ;
                2230            ; SUB 7C26
                2240            ; "INDCSR" PREDISPONE PB3,PB4
                2250            ; PER INDIRIZZARE IL "COMMAND
                2260            ; REGISTER" DEL 1771 (IN SCRIT=
                2270            ; TURA), OVVERO LO "STATUS RE=
                2280            ; GISTER (IN LETTURA).
                2290            ;
C085- BD 80 A8  2300 INDCSR     LDA DRB,X     ; PREDISPONE IDIRIZZAMENTO
C088- 29 E7     2310            AND #$E7      ; DEL REGISTRO COMMAND/STATUS
C08A- 4C 9C C0  2320            JMP EXESTA    ; (RTS)
                2330            ;
                2340            ; SUB 7C2E
                2350            ; "INDSR" PREDISPONE PB3,PB4
                2360            ; PER INDIRIZZARE IL SECTOR
                2370            ; REGISTER DEL 1771
                2380            ;
C08D- BD 80 A8  2390 INDSR      LDA DRB,X : PREDISP. INDIR. SECTOR REG.
C090- 29 F7     2400            AND #$F7
C092- 09 10     2410            ORA #$10
C094- 4C 9C C0  2420            JMP EXESTA    ; (RTS)
                2430            ;
                2440            ; SUB 7C38
                2450            ; "INDDR" PREDISPONE PB3,PB4
                2460            ; PER INDIRIZZARE IL "DATA
                2470            ; REGISTER" DEL 1771.
                2480            ;
C097- BD 80 A8  2490 INDDR      LDA DRB,X     ; PREDISP. INDIR. DATA REG.
C09A- 09 18     2500            ORA #$18
                2510            ;
C09C- 9D 80 A8  2520 EXESTA     STA DRB,X
C09F- 60        2530            RTS
                2540            ;
                2550            ; SUB 7C41
                2560            ; "EXECCOM" INVIA UN COMANDO
                2570            ; (DA METTERE PRIMA IN FDCCOM)
                2580            ; AL FDC 1771.
                2590            ;
C0A0- 20 6B C0  2600 EXECCOM    JSR SETWRITE          ; PREPARA SCRITT.
C0A3- 20 85 C0  2610            JSR INDCSR    ; INDIRIZZA COMMAND REG.
C0A6- A9 C8     2620            LDA #$C8
C0A8- 9D 89 A8  2630            STA T2CH,X    ; T2 CONTA PER 20 SEC. MAX.
C0AB- 20 B4 C0  2640            JSR DSYSIRQ   ; SETTA MASCHERA IRQ
C0AE- A5 04     2650            LDA *FDCCOM   ; RIPRENDE CODICE COMANDO
C0B0- 9D 81 A8  2660            STA DRA,X     ; LO SCRIVE NEL 1771 DA PORT A
C0B3- 60        2670            RTS
                2680            ;
                2690            ; SUB 7C56
                2700            ; "DSYSIRQ" DISABILITA L' IRQ
                2710            ; NELLA C.P.U. DEL SISTEMA, DI
                2720            ; MODO CHE L' IRQ EMESSO POI
                2730            ; DALLA VIA DELLA SCHEDA FDC
                2740            ; NON ABBIA EFFETTO ALCUNO.
                2750            ;
C0B4- 78        2760 DSYSIRQ    SEI  ; SETUP IRQ MASK
C0B5- BD 8D A8  2770            LDA IFR,X
C0B8- 9D 8D A8  2780            STA IFR,X     ; TRUCCO PER RESETTARE IFR
C0BB- A9 B2     2790            LDA #$B2
C0BD- 9D 8E A8  2800            STA IER,X     ; ABILITA IRQ DA T2, CB1, CA1
C0C0- 60        2810            RTS
                2820            ;
                2830            ; SUB 7C63
                2840            ; "IRQDIS" DISABILITA INVECE
                2850            ; OGNI POSSIBILITA' DI GENERA=
                2860            ; RE UN IRQ DALL' APPOSITO PIN
                2870            ; DELLA "VIA" SULLA SCHEDA FDC.
                2880            ; INOLTRE RIABILITA L' IRQ ALLA
                2890            ; CPU DEL SISTEMA.
                2900            ;
C0C1- A9 7F     2910 IRQDIS     LDA #$7F      ; DISABILITA IRQ DALLA VIA
C0C3- 9D 8E A8  2920            STA IER,X
C0C6- 58        2930            CLI  ; RIMUOVE MASCHERA IRQ
C0C7- 60        2940            RTS
                2950            ;
                2960            ; SUB 7C6A
                2970            ; "READSTATUS" FORZA LA
                2980            ; LETTURA DELLO STATO DAL
                2990            ; FDC 1771.
                3000            ;
C0C8- 20 78 C0  3010 READSTATUS JSR SETREAD   ; PREPARA LETTURA
C0CB- 20 85 C0  3020            JSR INDCSR    ; INDIRIZZA STATUS REG.
C0CE- 20 53 C0  3030            JSR FORCE     ; FORZA LA LETTURA
C0D1- BD 81 A8  3040            LDA DRA,X     ; STATO IN DRA
C0D4- 49 FF     3050            EOR #$FF      ; COMPLEMENTA
C0D6- 85 07     3060            STA *FDCSTATUS        ; SALVA
C0D8- 60        3070            RTS
                3080            ;
                3090            ; SUB 7C7C******
                3100            ; "CTRLBLOCK" E' LA ROUTINE
                3110            ; PRINCIPALE DEL NUCLEO DOS.
                3120            ; GESTISCE LA LETTURA, LA
                3130            ; SCRITTURA, LA VERIFICA DI
                3140            ; QUANTO E' LETTO O SCRITTO;
                3150            ; CONTROLLA CHE TUTTE LE OPERA=
                3160            ; ZIONI AVVENGANO CORRETTAMEN=
                3170            ; TE, E SE E' COSI' TORNA CON
                3180            ; $FF IN ERRCODE, ALTRIMENTI
                3190            ; RIEMPITA COL CODICE DI ERRORE
                3200            ; (CHE HA SEMPRE BIT 7 = 0).
                3210            ; LA ROUTINE E' RICURSIVA GRA=
                3220            ; ZIE ALL' USO OPPORTUNO DEI
                3230            ; PARAMETRI LA CUI LISTA E'
                3240            ; ALL' INIZIO DEL LISTATO.
                3250            ; OVVIAMENTE "CTRLBLOCK" GE=
                3260            ; STISCE IL CALCOLO DEI SET=
                3270            ; TORI/TRACCE SUCCESSIVI QUALORA
                3280            ; IL FILE ECCEDA I 128 BYTES.
                3290            ;
C0D9- A5 02     3300 CTRLBLOCK  LDA *SAVEPARM         ; RECUPERA CONTROLCODE
C0DB- C9 10     3310            CMP #INIT     ; RICHIESTA UNA INIZIALIZZAZIONE?
C0DD- F0 19     3320            BEQ INITENT   ; SI, SALTA
C0DF- C9 20     3330            CMP #DEINIT   ; E' UNA DE-INIZIALIZZAZIONE?
C0E1- F0 26     3340            BEQ DEINITENT         ; SI, SALTA
C0E3- C9 30     3350            CMP #INIT1    ; INIZI.NE DRIVER 1?
C0E5- D0 3B     3360            BNE DIVERSO   ; NO, SALTA
                3370            ;
C0E7- 20 4B C0  3380 INIT1ENT   JSR PUNTSCHED         ; *SETUP PER DRIVE 1*
C0EA- 20 0E C0  3390            JSR VIAINIT   ; INIZIALIZZA VIA
C0ED- 20 EF C2  3400            JSR VERIFY    ; VERIFICA CHE LA VIA E' OK
C0F0- B0 03     3410            BCS INIT1RET          ; C=1 SE SCHEDA NON OK
C0F2- 20 03 C0  3420            JSR SETDRIVE1         ; PREPARA IND, DRIVER 1
C0F5- 4C 82 C1  3430 INIT1RET   JMP BLKRET    ; TORNA, EVENTUALMENTE CON C=1
                3440            ;
C0F8- 20 4B C0  3450 INITENT    JSR PUNTSCHED         ; *SETUP PER DRIVE 2*
C0FB- 20 0E C0  3460            JSR VIAINIT   ; INIZIALIZZA VIA
C0FE- 20 EF C2  3470            JSR VERIFY    ; VERIFICA SIA OK
C101- B0 03     3480            BCS INITRET   ; C=1 SE SCHEDA NON OK
C103- 20 06 C0  3490            JSR SETDRIVE2         ; PREPARA IND. DRIVER 2
C106- 4C 82 C1  3500 INITRET    JMP BLKRET    ; TORNA (EVEN.TE CON C=1)
                3510            ;
C109- 20 4B C0  3520 DEINITENT  JSR PUNTSCHED
C10C- 20 EF C2  3530            JSR VERIFY    ; VERIFICA LA SCHEDA COMUNQUE
C10F- B0 0E     3540            BCS DEINITRET         ; ESCE CON C=1 SE ERR.
C111- 20 09 C0  3550            JSR DRIVEOFF          ; PREPARA PER MOTOR OFF
C114- 20 61 C0  3560            JSR MOTOR     ; DISATTIVA DRIVER
C117- 20 89 C2  3570            JSR TRINDR    ; FORZA (TR) IN DR DEL F.D.C.
C11A- A9 EF     3580            LDA #$EF      ; "SEEK/NO VERIFY" COMMAND
C11C- 20 BA C2  3590            JSR SEEKSTOP          ; SEEK CON DISCO FERMO!
C11F- 4C 82 C1  3600 DEINITRET  JMP BLKRET    ; TORNA, EVENT.TE CON C=1 SE ERR.
                3610            ;
C122- A5 0B     3620 DIVERSO    LDA *INITADDR         ; INDIR. INIZ. NELLA BASE
C124- 85 13     3630            STA *BASE+2
C126- A5 0C     3640            LDA *INITADDR+1
C128- 85 14     3650            STA *BASE+3
C12A- A9 00     3660            LDA #0
C12C- 85 10     3670            STA *NERR     ; AZZERA N. ERRORI LETTURA
C12E- A9 1E     3680            LDA #$1E      ; 30 = N. TENTATIVI
C130- 85 0F     3690            STA *NTENT
                3700            ;
C132- 20 4B C0  3710 LOOP1      JSR PUNTSCHED
C135- 20 EF C2  3720            JSR VERIFY    ; VERIFICA IN OGNI CASO
C138- B0 48     3730            BCS BLKRET    ; ESCE SE ERRORE (C=1)
C13A- 20 0A C2  3740            JSR DRIVCTRL          ; DRIVER SETUP OK?
C13D- B0 43     3750            BCS BLKRET    ; ESCE SE ERRORE
C13F- 20 AB C2  3760 LOOP7      JSR SEEK      ; SEEK&VERIFY CON DISCO RUOT.
C142- B0 3E     3770            BCS BLKRET    ; ESCE SE ERRORE
C144- 20 E0 C2  3780 LOOP6      JSR NSECINSR          ; N. SETT. IN S.REG.
C147- 20 05 C3  3790            JSR SETADDR   ; AGGIORNA BASE 128 A 128
C14A- A5 02     3800            LDA *SAVEPARM         ; RIPRENDE PARAM.
C14C- F0 3E     3810            BEQ READ/CENT         ; SALTA SE LETT.+VERIF.
C14E- C9 01     3820            CMP #WRITCOMP
C150- F0 3D     3830            BEQ WRIT/CENT         ; SALTA SE SCRITT.+VERIF.
C152- C9 03     3840            CMP #READ
C154- F0 0F     3850            BEQ READENT   ; SALTA SE: "SOLA LETTURA"
C156- C9 04     3860            CMP #WRITE
C158- F0 19     3870            BEQ WRITENT   ; SALTA SE: "SOLA SCRITTURA"
C15A- C9 06     3880            CMP #COMPARE
C15C- F0 51     3890            BEQ COMPENT   ; SALTA SE "SOLA VERIFICA"
C15E- A9 17     3900            LDA #$17      ; ERRCODE: PARAM. NON AMMESSO
C160- 85 05     3910            STA *ERRCODE          ; SALVA CODICE ERRORE
C162- 38        3920            SEC  ; C=1 PER RITORNO DA ERRORE
C163- B0 1D     3930            BCS BLKRET    ; BRANCH ALWAYS SE ERRORE
                3940            ;
C165- 20 1A C3  3950 READENT    JSR READSECT          ; READ SECTOR!
C168- B0 18     3960            BCS BLKRET    ; ESCE CON ERRORE SE C=1
C16A- 20 C8 C0  3970            JSR READSTATUS        ; LETTURA STATO F.D.C.
C16D- 29 9F     3980            AND #$9F      ; ESTRAE NOTREADY,NOTFOUND,CRC,
C16F- F0 60     3990            BEQ LOOP2     ; LOSTDATA, DRQ, BUSY :SALAT SE OK
C171- D0 0A     4000            BNE STATERR   ; SALTA: STATUS ERROR!
```

I floppy disk controller: teoria e pratica di progetto hard/soft

Seguito listato.

```
                 4010                ;
C173- 20 5B C3   4020 WRITENT       JSR WRITSECT          ; WRITE SECTOR!
C176- B0 0A      4030               BCS BLKRET   ; ESCE SE ERRORE
C178- 20 C8 C0   4040               JSR READSTATUS        ; LETTURA STATO F.D.C.
C17B- F0 54      4050               BEQ LOOP2    ; TUTTO OK:PROSEGUE
C17D- 20 BA C3   4060 STATERR       JSR RETRY    ; RITENTA LETTURA 30 VOLTE
C180- 90 B0      4070 BRLOOP1       BCC LOOP1    ; RILETTURA OK:NUOVO SETTORE!
                 4080                ;    ... ALTRIMENTI ....
C182- 20 C1 C0   4090 BLKRET        JSR IRQDIS   ; DISABILITA IRQ NELLA VIA
C185- B0 04      4100               BCS BLKRET1  ; ESCE SE ERRORE (C=1)
C187- A9 FF      4110               LDA #$FF     ; CODICE DI "NOT ERROR"
C189- 85 05      4120               STA *ERRCODE          ; LO SALVA
C18B- 60         4130 BLKRET1       RTS  ; FINE PROCEDURA!
                 4140                ;
C18C- A9 03      4150 READ/CENT     LDA #03      ; PARAM. PER RICURSIVITA'
C18E- 2C         4160               .BY $2C      ; TRUCCO FALSO "BIT"
C18F- A9 04      4170 WRIT/CENT     LDA #04      ; PARAM. PER RICURSIVITA'
C191- 85 02      4180               STA *SAVEPARM         ; SALVA PER DOPO
C193- A5 08      4190               LDA *NTRACK
C195- 48         4200               PHA
C196- A5 09      4210               LDA *NSECTOR
C198- 48         4220               PHA  ; HA SALVATO TRACK&SECTOR
C199- 20 32 C1   4230               JSR LOOP1    ; RICURSIVITA' IN ATTO!
C19C- 90 04      4240               BCC OKRET    ; EXIT OK!
C19E- 68         4250               PLA  ; ALTRIMENTI RESETTA STACK
C19F- 68         4260               PLA
C1A0- B0 E0      4270               BCS BLKRET   ; ESCE CON C=1
                 4280                ;
C1A2- A9 06      4290 OKRET         LDA #COMPARE          ; VERIFICA DOPO RD O WR
C1A4- 85 02      4300               STA *SAVEPARM         ; SALVA
C1A6- 68         4310               PLA  ; RECUPERA TRACK&SECTOR
C1A7- 85 09      4320               STA *NSECTOR
C1A9- 68         4330               PLA
C1AA- 85 08      4340               STA *NTRACK
C1AC- 4C D9 C0   4350               JMP CTRLBLOCK         ; TORNA ALL' INIZIO
                 4360                ;
C1AF- 20 86 C3   4370 COMPENT       JSR COMPSECT          ; CONFRONTA SETTORE!
C1B2- B0 CE      4380               BCS BLKRET   ; ESCE SE ERRORE
C1B4- 20 C8 C0   4390               JSR READSTATUS        ; LEGGE STATO F.D.C.
C1B7- 29 9F      4400               AND #$9F     ; NRDY,NFND,CRC,LOST,DRQ,BSY
C1B9- D0 0B      4410               BNE CSTATERR          ; SALTA SE ERRORE DI STATO
C1BB- A5 10      4420               LDA *NERR    ; ERRORI DI LETTURA?
C1BD- F0 12      4430               BEQ LOOP2    ; NO!
C1BF- A9 23      4440               LDA #$23     ; COMPARE FAULT STATUS ERR.
C1C1- 85 05      4450               STA *ERRCODE          ; SALVA COD. ERR.
C1C3- 38         4460               SEC  ; C=1 PER ERROR
C1C4- B0 BC      4470               BCS BLKRET   ; BRANCH ALWAYS
                 4480                ;
C1C6- A9 00      4490 CSTATERR      LDA #0       ; SE ERRORE DI STATO...
C1C8- 85 10      4500               STA *NERR    ; AZZERA N. ERRORI ...
C1CA- 20 BA C3   4510               JSR RETRY    ; ...POI RITENTA!
C1CD- B0 B3      4520               BCS BLKRET   ; ESCE SE NULLA DA FARE
C1CF- 90 AF      4530               BCC BRLOOP1  ; (JMP LOOP1) NEW SECT.
C1D1- A9 1E      4540 LOOP2         LDA #$1E     ; RESET N. TENTAT.
C1D3- 85 0F      4550               STA *NTENT
C1D5- 18         4560               CLC  ; PREPARA CONFINE NUOVO SETT.
C1D6- A5 13      4570               LDA *BASE+2
C1D8- 65 0A      4580               ADC *CONF
C1DA- 85 13      4590               STA *BASE+2
C1DC- A5 14      4600               LDA *BASE+3
C1DE- 69 00      4610               ADC #0
C1E0- 85 14      4620               STA *BASE+3
C1E2- C5 0E      4630               CMP *ENDADDR+1
C1E4- D0 0A      4640               BNE LOOP3
C1E6- A5 13      4650               LDA *BASE+2
C1E8- C5 0D      4660               CMP *ENDADDR          ; FINE?
C1EA- D0 04      4670               BNE LOOP3
C1EC- 18         4680               CLC  ; SI, ESCE DALLA PROCEDURA
C1ED- 4C 82 C1   4690               JMP BLKRET
                 4700                ;
C1F0- E6 13      4710 LOOP3         INC *BASE+2
C1F2- D0 02      4720               BNE LOOP4
C1F4- E6 14      4730               INC *BASE+3
C1F6- A9 10      4740 LOOP4         LDA #$10     ; SEDICESIMO SETTORE?
C1F8- C5 09      4750               CMP *NSECTOR
C1FA- F0 05      4760               BEQ LOOP5    ; SI, SALTA
C1FC- E6 09      4770               INC *NSECTOR
C1FE- 4C 44 C1   4780               JMP LOOP6    ; RICOMINCIA (SENZA SEEK)
                 4790                ;
C201- A9 01      4800 LOOP5         LDA #01      ; PRIMO SETTORE NUOVA TR.
C203- 85 09      4810               STA *NSECTOR
C205- E6 08      4820               INC *NTRACK  ; AGGIORNA N.TR.
C207- 4C 3F C1   4830               JMP LOOP7    ; RICOMINCIA (CON SEEK)
                 4840                ;
                 4850                ;
                 4860                ; SUB 7DC5
                 4870                ; "DRIVCTRL" CONTROLLA LO STATO
                 4880                ; DEL DRIVER: SE GIA' GIRA O SE
                 4890                ; -DA FERMO- DEVE INIZIARE
                 4900                ; A GIRARE; INOLTRE (7DFF)
                 4910                ; VERIFICA CHE IL DISCO SIA IN=
                 4920                ; SERITO NEL DRIVER E CHE STIA
                 4930                ; GIRANDO EFFETTIVAMENTE.
                 4940                ;
C20A- A5 06      4950 DRIVCTRL      LDA *NDRIVER          ; VERIFICA CHE NDRIVER
C20C- 29 06      4960               AND #6       ; ..SIA GIA' OK PER..
C20E- C9 06      4970               CMP #6       ; ..LA SELEZ. DRIV. 1 O 2
C210- D0 04      4980               BNE DCTR1    ; SI, DIVERSA DA 6 (MOT.OFF)
C212- A9 02      4990               LDA #2       ; NOT-READY ERR. CODE
C214- D0 26      5000               BNE DCTR2    ; ESCE CON ERR.
C216- BD 80 A8   5010 DCTR1         LDA DRB,X    ; VEDE SITUAZ. LINEE
C219- 29 06      5020               AND #6       ; ESTRAE DRA0,1 E ...
C21B- 85 0A      5030               STA *DRSTATUS         ; PER ORA SALVA..
C21D- 20 61 C0   5040               JSR MOTOR    ; ATTIVA I MOTORI
C220- A5 0A      5050               LDA *DRSTATUS         ; GUARDA SE E' LA PRIMA...
C222- C9 06      5060               CMP #6       ; ..VOLTA CHE ACCENDE MOTORI:
C224- D0 1A      5070               BNE DCTR3    ; SALTA:MOT.GIA' ACCESO
C226- 20 C1 C0   5080               JSR IRQDIS   ; SE INVECE I MOTORI..
C229- A9 0A      5090               LDA #$A      ; ..STANNO AVVIANDOSI...
C22B- 9D 89 A8   5100               STA T2CH,X   ; ..LASCIA UN TRANSITORIO...
C22E- 20 B4 C0   5110               JSR DSYSIRQ  ; ..DI AVVIAMENTO (1 SEC.)...
C231- BD 8D A8   5120 DCTR4         LDA IFR,X    ; ATTENDE IRQ...
C234- 10 FB      5130               BPL DCTR4    ; ..FINCHE' NON ARRIVA
C236- 29 20      5140               AND #$20     ; IRQ DA TIME-OUT?
C238- D0 06      5150               BNE DCTR3    ; SI, STARTUP FINITO
C23A- A9 12      5160               LDA #$12     ; "UNWANTED IRQ" ERR. CODE
C23C- 85 05      5170 DCTR2         STA *ERRCODE          ; LO SALVA
C23E- 38         5180               SEC  ; C=1 PER USCIRE CON ERR.
C23F- 60         5190 DCTRET        RTS
                 5200                ;
C240- 20 52 C2   5210 DCTR3         JSR DISCVER  ; VERIFICA CHE VI SIA IL...
C243- B0 FA      5220               BCS DCTRET   ; ..DISCO E GIRI (C=1 SE ERR.)
C245- A5 06      5230               LDA *NDRIVER          ; SE PRIMA LA SITUAZIONE...
C247- 29 06      5240               AND #6       ; ..DELLE LINEE DI CONTROLLO...
C249- C5 0A      5250               CMP *DRSTATUS         ; ..ERA EGUALE..
C24B- F0 3A      5260               BEQ DCTRET1  ; ..FINE:SE NO...
C24D- A9 F7      5270               LDA #$F7     ; ..ESEGUE RESTORE!
C24F- 4C BA C2   5280               JMP EXECCOM1          ; (JSR-RTS)
                 5290                ;
C252- 20 89 C2   5300 DISCVER       JSR TRINDR   ; SEEK TRACCIA ATTUALE
C255- A9 E7      5310               LDA #$E7     ; SEEK CODE
C257- 20 BA C2   5320               JSR SEEKSTOP
C25A- B0 E3      5330               BCS DCTRET   ; C=1 SE ERRORE DI SEEK
C25C- 20 C1 C0   5340               JSR IRQDIS   ; DISAB. VIA IRQ
C25F- A9 03      5350               LDA #3       ; 250 m.SEC. SETUP
C261- 9D 89 A8   5360               STA T2CH,X   ; ..PER FARE FARE AL...
C264- 20 B4 C0   5370               JSR DSYSIRQ  ; ..DISCO ALMENO 1 GIRO..
C267- 20 C8 C0   5380 DVER2         JSR READSTATUS        ; ..LEGGE STATO FDC..
C26A- 29 02      5390               AND #2       ; ESTRAE SOLITI FLAGS..
C26C- D0 09      5400               BNE DVER1    ; SALTA, TUTTO OK
C26E- BD 8D A8   5410               LDA IFR,X    ; SE NO RITENTA ALMENO...
C271- 10 F4      5420               BPL DVER2    ; ..FINO A TIMEOUT
C273- A9 14      5430               LDA #20      ; "DISK NOT ROTATE" ERR.
C275- D0 C5      5440               BNE DCTR2    ; ESCE CON L' ERR.
                 5450                ;
C277- 20 C8 C0   5460 DVER1         JSR READSTATUS        ; RILEGGE STATO F.D.C.
C27A- 29 02      5470               AND #2       ; ADESSO FLAG "INDEX" ...
C27C- F0 09      5480               BEQ DCTRET1  ; ..E' DIVERSO SE C'E' DISCO
C27E- BD 8D A8   5490               LDA IFR,X    ; ..SE NO, RILEGGE STATO..
C281- 10 F4      5500               BPL DVER1    ; ..FINO A TIMEOUT.
C283- A9 21      5510               LDA #$21     ; "NO DISKETTE" ERROR
C285- D0 B5      5520               BNE DCTR2    ; ESCE CON ERRORE
C287- 18         5530 DCTRET1       CLC  ; TUTTO OK,
C288- 60         5540 DVERRET       RTS  ; ESCE
                 5550                ;
                 5560                ; SUB 7E4A
                 5570                ; "TRINDR" FORZA NEL DATA REGISTER
                 5580                ; L' ATTUALE CONTENUTO DEL TRACK REG.
                 5590                ;
C289- 20 78 C0   5600 TRINDR        JSR SETREAD  ; PREPARA LETTURA
C28C- 20 A0 C2   5610               JSR INDTR    ; PREDISP. IND. TRACK REG.
C28F- 20 53 C0   5620               JSR FORCE    ; FORZA LETTURA
C292- BC 81 A8   5630               LDY DRA,X    ; (Y IN A, A IN DR)
C295- 20 97 C0   5640               JSR INDDR    ; PREDISP. IND. DATA REG.
C298- 20 6B C0   5650               JSR SETWRITE          ; PREPARA LETTURA
C29B- 98         5660               TYA
C29C- 9D 81 A8   5670               STA DRA,X    ; SCRIVE
C29F- 60         5680               RTS  ; ORA IN DR C' E' (TR)
                 5690                ;
                 5700                ; SUB 7E61
                 5710                ; "INDTR" PREDISPONE PB3,4 PER
                 5720                ; INDIRIZZARE IL TRACK REGISTER
                 5730                ;
C2A0- BD 80 A8   5740 INDTR         LDA DRB,X    ; PREDISPONE PER
C2A3- 29 EF      5750               AND #$EF     ; INDIRIZZARE IL T.REG.
C2A5- 09 08      5760               ORA #8       ; DEL F.D.C.
C2A7- 9D 80 A8   5770               STA DRB,X
C2AA- 60         5780               RTS
                 5790                ;
                 5800                ; SUB 7E6C/7E7C
                 5810                ; "SEEK" ESEGUE UN "SEEK" DEL=
                 5820                ; LA TRACCIA DESCRITTA IN "NTRACK",
                 5830                ; CON HEADLOAD & VERIFY; IN "SEEK=
                 5840                ; STOP" INVECE SI ENTRA CON IL CODICE
                 5850                ; DI SEEK SENZA HL NE' VERIFY QUANDO
                 5860                ; SI VUOLE FORZARE UNA AUTOVERIFICA
                 5870                ; NEL CHIP FDC; INOLTRE EXECCOM1 (CHE
                 5880                ; E' LO STESSO DI SEEKSTOP) E' IL PUNTO
                 5890                ; DI INGRESSO PER ESEGUIRE UN QUALSIASI
                 5900                ; COMANDO (DESCRITTO IN FDCCOM) IN MODO
                 5910                ; VERAMENTE COMPLETO, IN QUANTO VIENE
                 5920                ; ANCHE ATTESA LA FINE FISICA (DA "INTRQ")
                 5930                ; E VIENE SUBITO LETTO LO STATO DEL FDC.
                 5940                ;
C2AB- 20 97 C0   5950 SEEK          JSR INDDR    ; INDIR. DATA REG.
C2AE- 20 6B C0   5960               JSR SETWRITE          ; PREPARA SCRITTURA
C2B1- A5 08      5970               LDA *NTRACK  ; PRENDE No. TRACK
C2B3- 49 FF      5980               EOR #$FF     ; COMPLEMENTA
C2B5- 9D 81 A8   5990               STA DRA,X    ; LO SCRIVE (IN D.REG.)
C2B8- A9 E3      6000               LDA #$E3     ; SEEK, HEADLOAD, VERIFY
                 6010 EXECCOM1
C2BA- 85 04      6020 SEEKSTOP      STA *FDCCOM  ; SALVA COD. COMANDO
C2BC- 20 A0 C0   6030               JSR EXECCOM  ; ESEGUE COMANDO
C2BF- BD 8D A8   6040 SEEK1         LDA IFR,X    ; ATTENDE FINE ESECUZIONE
C2C2- 10 FB      6050               BPL SEEK1    ; (DA LINEA "INTRQ")
C2C4- 29 10      6060               AND #$10     ; INTRQ?
C2C6- F0 0D      6070               BEQ SEEK2    ; NO,SALTA
C2C8- A9 1A      6080               LDA #$1A     ; IN OGNI CASO PREP. ERRCODE
C2CA- 85 05      6090               STA *ERRCODE
C2CC- 20 C8 C0   6100               JSR READSTATUS        ; LEGGE STATO F.D.C.
C2CF- 29 19      6110               AND #$19     ; CRC,SEEK,BUSY
C2D1- D0 0B      6120               BNE SEEKERR  ; ERRORE, ESCE!
C2D3- 18         6130               CLC  ; TUTTO OK,
C2D4- 60         6140               RTS  ; TORNA CON C=0
                 6150                ;
C2D5- A9 12      6160 SEEK2         LDA #$12     ; UNWANTED IRQ ERRCODE
C2D7- 85 05      6170               STA *ERRCODE
C2D9- BD 8D A8   6180               LDA IFR,X    ; RICORDA LA SITUAZIONE
C2DC- 85 07      6190               STA *FDCSTATUS        ; PER EVENTUALE AS=
C2DE- 38         6200 SEEKERR       SEC  ; =SISTENZA DOPO
C2DF- 60         6210               RTS  ; E TORNA (CON ERRORE)
                 6220                ;
                 6230                ; SUB 7EA6
                 6240                ; "NSECINSR" SCRIVE IL NUMERO
                 6250                ; DEL SETTORE (DA "NSECTOR") NEL
                 6260                ; SECTOR REGISTER DEL FDC.
                 6270                ;
C2E0- 20 8D C0   6280 NSECINSR      JSR INDSR    ; PREPARA IND. S.REG.
C2E3- 20 6B C0   6290               JSR SETWRITE          ; PREPARA SCRITT.
C2E6- A5 09      6300               LDA *NSECTOR          ; PRENDE N.SETT.
C2E8- 49 FF      6310               EOR #$FF     ; COMPLEMENTA
C2EA- 9D 81 A8   6320               STA DRA,X    ; SCRIVE (NEL F.D.C)
C2ED- 18         6330               CLC
C2EE- 60         6340               RTS
                 6350                ;
                 6360                ; SUB 7EB6
                 6370                ; "VERIFY" CONTROLLA LA CORRETTA
                 6380                ; INIZIALIZZAZIONE DELLA SCHEDA FDC,
                 6390                ; BASANDOSI SU QUELLE INIZIALIZZAZIO=
                 6400                ; NI CHE SONO SEMPRE PRESENTI E FISSE,
                 6410                ; CIOE' L' ACR ED IL DDRB. TORNA CON
                 6420                ; C=1 SE ERRORE.
                 6430                ;
C2EF- A9 E0      6440 VERIFY        LDA #$E0     ; MASCHERA PER ACR
C2F1- DD 8B A8   6450               CMP ACR,X    ; ACR OK?
C2F4- D0 09      6460               BNE VER1     ; NO! ERRORE
C2F6- A9 BF      6470               LDA #$BF     ; MASCHERA PER PORTB
C2F8- DD 82 A8   6480               CMP DDRB,X   ; PORTB OK?
C2FB- D0 02      6490               BNE VER1     ; NO! ERRORE
C2FD- 18         6500               CLC  ; OK, TUTTO BENE
C2FE- 60         6510               RTS  ; TORNA CON C=0
C2FF- A9 1F      6520 VER1          LDA #$1F     ; "NO CONTROLLER" ERRCODE
C301- 85 05      6530               STA *ERRCODE
C303- 38         6540               SEC  ; C=1 PER ERRORE
C304- 60         6550               RTS  ; TORNA
                 6560                ;
                 6570                ; SUB 7ECD
                 6580                ; "SETADDR" LAVORA SULL' IN=
                 6590                ; DIRIZZO DELLA ZONA DI RAM
```

Seguito listato.

```
                6600            ; INTERESSATA, PER AVANZARE
                6610            ; CORRETTAMENTE DI 128 IN 128 BYTE.
                6620            ;
C305- 38        6630 SETADDR    SEC
C306- A5 0D     6640            LDA *ENDADDR          ; PARTE BASSA FINE
C308- E5 13     6650            SBC *BASE+2
C30A- 30 09     6660            BMI STAD1
C30C- 85 0A     6670            STA *CONF    ; SALVA TEMP.
C30E- A8        6680            TAY
C30F- A5 0E     6690            LDA *ENDADDR+1
C311- E5 14     6700            SBC *BASE+3  ; CALC. PARTE ALTA
C313- F0 04     6710            BEQ STADRET
C315- A0 7F     6720 STAD1      LDY #$7F     ; 128 BYTE
C317- 84 0A     6730            STY *CONF
C319- 60        6740 STADRET    RTS
                6750            ;
                6760            ; SUB 7EE6
                6770            ; "READSECT" E' LA ROUTINE CHE
                6780            ; PROVVEDE ALLA LETTURA DA DISCO
                6790            ; SCRIVENDO, QUANTO E' LETTO, NELLA
                6800            ; ZONA DI RAM FRA INITADDR ED ENDADDR
                6810            ;
C31A- A9 77     6820 READSECT   LDA #$77     ; READ COMMAND!
C31C- 85 04     6830            STA *FDCCOM  ; SALVA
C31E- 20 A0 C0  6840            JSR EXECCOM  ; COMANDA "READ SECTOR"
C321- 20 97 C0  6850            JSR INDDR    ; PREPARA IND. D.REG.
C324- 20 78 C0  6860            JSR SETREAD  ; PREPARA LETTURA
C327- BD 8D A8  6870 RSEC1      LDA IFR,X    ; DATO PRONTO?
C32A- 10 FB     6880            BPL RSEC1    ; NO, ATTENDE
C32C- 29 02     6890            AND #2       ; E' IL "DRQ"?
C32E- F0 17     6900            BEQ RSEC3    ; NO, SALTA!
C330- BD 81 A8  6910            LDA DRA,X    ; SI, LEGGI IL DATO
C333- 91 13     6920            STA (BASE+2),Y      ; SCRIVILO IN RAM
C335- 88        6930            DEY  ; DECR. CONTO 128 BYTES
C336- 10 EF     6940            BPL RSEC1    ; ASPETTA DATO SEGUENTE
C338- BD 8D A8  6950 RSEC2      LDA IFR,X    ; Y)$7F:FINE ESECUZ.?
C33B- 10 FB     6960            BPL RSEC2    ; ATTENDE
C33D- 29 02     6970            AND #2       ; E' ULTIMO "DRQ"?
C33F- F0 06     6980            BEQ RSEC3    ; NO, SALTA
C341- BD 81 A8  6990            LDA DRA,X    ; SI, LETTURA "DUMMY"...
C344- 4C 38 C3  7000            JMP RSEC2    ; ..PER "SCARICARE" FDC
C347- BD 8D A8  7010 RSEC3      LDA IFR,X    ; INTRQ?
C34A- 29 10     7020            AND #$10
C34C- F0 02     7030            BEQ RSEC5    ; NO, ERRORE!
C34E- 18        7040            CLC  ; SI, FINE LETTURA SETTORE OK
C34F- 60        7050            RTS  ; TORNA CON C=0
                7060            ;
C350- A9 51     7070 RSEC5      LDA #$51     ; "TIME OUT" ERRCODE
C352- 85 05     7080            STA *ERRCODE
C354- BD 8D A8  7090            LDA IFR,X    ; IN TALE CASO SALVA LA..
C357- 85 07     7100            STA *FDCSTATUS      ; ..SITUAZIONE PER..
C359- 38        7110            SEC  ; ..EVENTUALI ANALISI..
C35A- 60        7120            RTS  ; TORNA CON ERR. (C=1)
                7130            ;
                7140            ; SUB 7F2A
                7150            ; "WRITSECT" E' LA ROUTINE CHE
                7160            ; PROVVEDE ALLA SCRITTURA SU DISCO
                7170            ; DEI DATI PRESI DALLA RAM FRA INITADDR
                7180            ; ED ENDADDR
                7190            ;
C35B- A9 57     7200 WRITSECT   LDA #$57     ; WRITE SECTOR CODE
C35D- 85 04     7210            STA *FDCCOM  ; SALVA
C35F- 20 A0 C0  7220            JSR EXECCOM  ; COMANDA SCRITTURA SETT.
C362- 20 97 C0  7230            JSR INDDR    ; PREPARA IND. DATA R.
                7240            ; NOTA : GIA' PREPARATO PER SCRITTURA!
C365- BD 8D A8  7250 WSEC1      LDA IFR,X    ; DATO PRONTO?
C368- 10 FB     7260            BPL WSEC1    ; NO, ASPETTA
C36A- 29 02     7270            AND #2       ; E' PROPRIO IL "DRQ"?
C36C- F0 D9     7280            BEQ RSEC3    ; NO, SALTA
C36E- B1 13     7290            LDA (BASE+2),Y      ; PRENDI DATO DA RAM
C370- 9D 81 A8  7300            STA DRA,X    ; SCRIVILO IN F.D.C.
C373- 88        7310            DEY  ; DECR. CONTO 128 BYTES
C374- 10 EF     7320            BPL WSEC1    ; FINITI I 128?
C376- BD 8D A8  7330 WSEC2      LDA IFR,X    ; DRQ ANCORA?
C379- 10 FB     7340            BPL WSEC2    ; ATTENDE..
C37B- 29 02     7350            AND #2       ; SE NON E' DRQ SALTA
C37D- F0 C8     7360            BEQ RSEC3    ; E ATTENDERA' INTRQ
C37F- A9 00     7370            LDA #0       ; SE NO, SCRITTURA ...
C381- 9D 81 A8  7380            STA DRA,X    ; ..DUMMY PER "SCA=
C384- F0 F0     7390            BEQ WSEC2    ; RICARE" IL F.D.C.
                7400            ;
                7410            ; SUB 7F56
                7420            ; "COMPSECT" PROVVEDE A LEGGERE
                7430            ; DA DISCO, CONFRONTANDO BYTE A
                7440            ; BYTE COL CONTENUTO DELLA RAM FRA
                7450            ; INITADDR ED ENDADDR.
                7460            ;
C386- A9 77     7470 COMPSECT   LDA #$77     ; READ SECTOR CODE
C388- 85 04     7480            STA *FDCCOM
C38A- 20 A0 C0  7490            JSR EXECCOM  ; COMANDA LETTURA SETT.
C38D- 20 97 C0  7500            JSR INDDR    ; PREP. INDIR. DATA R.
C390- 20 78 C0  7510            JSR SETREAD  ; PREPARA LETTURA
C393- BD 8D A8  7520 CSEC0      LDA IFR,X    ; DATO PRONTO DAL FDC?
C396- 10 FB     7530            BPL CSEC0    ; ATTENDE!
C398- 29 02     7540            AND #2       ; E' PROPRIO DRQ?
C39A- F0 AB     7550 CSEC1      BEQ RSEC3    ; NO, SALTA
C39C- BD 81 A8  7560            LDA DRA,X    ; LEGGI IL DATO...
C39F- D1 13     7570            CMP (BASE+2),Y      ; ...E CONFRONTA CON RAM
C3A1- D0 12     7580            BNE CSEC3    ; ERRORE! SALTA
C3A3- 88        7590            DEY  ; DECR. CONTO 128 SE E' OK
C3A4- 10 ED     7600            BPL CSEC0    ; NUOVO DATO
C3A6- BD 8D A8  7610 CSEC2      LDA IFR,X    ; NUOVO IRQ?
C3A9- 10 FB     7620            BPL CSEC2    ; ATTENDE
C3AB- 29 02     7630            AND #2       ; DRQ?
C3AD- F0 98     7640            BEQ RSEC3    ; NO, SALTA
C3AF- BD 81 A8  7650            LDA DRA,X    ; SI, LETTURA DUMMY
C3B2- 4C A6 C3  7660            JMP CSEC2    ; E RICOMINCIA
C3B5- E6 10     7670 CSEC3      INC *NERR    ; SE CMP (),SEGNA ERR.
C3B7- 4C A6 C3  7680            JMP CSEC2    ; E PROSEGUE
                7690            ;
                7700            ; SUB 7F8C
                7710            ; "RETRY" FA SI' CHE SI RITENTI
                7720            ; PER BEN 30 VOLTE LA LETTURA: DOPO
                7730            ; 10 TENTATIVI PROVA A MUOVERE LA TE=
                7740            ; STINA SULLA TRACCIA SUCCESSIVA, LA
                7750            ; RIPORTA DI NUOVO SULLA TRACCIA DA
                7760            ; LEGGERE E TENTA ALTRE 10 VOLTE, DOPO=
                7770            ; DICHE' PROVA A PORTARE LA TESTINA
                7780            ; SULLA TRACCIA PRECEDENTE E DI NUOVO
                7790            ; SULLA TRACCIA DA LEGGERE. SE DOPO GLI
                7800            ; ULTIMI 10 TENTATIVI IL RISULTATO E'
                7810            ; ANCORA NEGATIVO, TORNA CON IL CODICE
                7820            ; DI ERRORE "DATO NON PIU' RILEGGIBILE".
                7830            ;
C3BA- A9 22     7840 RETRY      LDA #$22     ; COMPARE FAULT ERR.
C3BC- 85 05     7850            STA *ERRCODE        ; SALVA
C3BE- C6 0F     7860            DEC *NTENT
C3C0- F0 31     7870            BEQ RETRY5   ; FINE 30 TENTATIVI
C3C2- A5 0F     7880            LDA *NTENT   ; QUANTI TENT. FATTI?
C3C4- C9 0A     7890            CMP #10
C3C6- F0 06     7900            BEQ RETRY2   ; GIA' TENTATO 20 VOLTE!
C3C8- C9 14     7910            CMP #20
C3CA- F0 13     7920            BEQ RETRY3   ; GIA' TENTATO 10 VOLTE!
C3CC- 18        7930 RETRY1     CLC  ; TENTATIVI OK!
C3CD- 60        7940            RTS  ; TORNA CON C=0
                7950            ;
C3CE- A5 08     7960 RETRY2     LDA *NTRACK  ; E' SU TRACC. 0?
C3D0- C9 00     7970            CMP #0
C3D2- F0 F8     7980            BEQ RETRY1   ; SI, FINE TENT.
C3D4- A9 A0     7990            LDA #$A0     ; STEP IN COMMAND!
C3D6- 20 BA C2  8000            JSR EXECCOM1        ; ESEGUE STEP IN
C3D9- B0 18     8010            BCS RETRY5   ; ERRORE:ESCE!
C3DB- A9 80     8020            LDA #$80     ; STEP OUT COMMAND!
C3DD- D0 0F     8030            BNE RETRY4   ; ESEGUE STEP OUT
C3DF- A5 08     8040 RETRY3     LDA *NTRACK  ; E' SU TRACK 41?
C3E1- C9 29     8050            CMP #41      ; E' SULLA TRACC. 41?
C3E3- F0 E7     8060            BEQ RETRY1   ; SI, TORNA
C3E5- A9 80     8070            LDA #$80     ; NO, STEP OUT!
C3E7- 20 BA C2  8080            JSR EXECCOM1        ; ESEGUE STEPOUT
C3EA- B0 07     8090            BCS RETRY5   ; ESCE SE ERRORE
C3EC- A9 A0     8100            LDA #$A0     ; STEP IN COMMAND
C3EE- 20 BA C2  8110 RETRY4     JSR EXECCOM1        ; ESEGUE STEP IN
C3F1- 90 D9     8120            BCC RETRY1   ; TORNA SE TUTTO OK
C3F3- A9 1D     8130 RETRY5     LDA #$1D     ; "IRRECOVERABLE DATA"
C3F5- 85 05     8140            STA *ERRCODE
C3F7- 38        8150            SEC  ; C=1 PER ERRORE
C3F8- 60        8160            RTS
                8170            ;
                8180            ;
                8190            ; PROGRAMMA PRINCIPALE
                8200            ;
                8210            ;
C3F9- D8        8220 START1     CLD
C3FA- A5 01     8230            LDA *NDISK
C3FC- C9 01     8240            CMP #1
C3FE- F0 03     8250            BEQ DISK1
C400- A9 10     8260            LDA #INIT    ; INIZIALIZZA PER DRIV. 2
C402- 2C        8270            .BY $2C
C403- A9 30     8280 DISK1      LDA #INIT1   ; INIZIAL. DRIV. 1
C405- 85 02     8290            STA *SAVEPARM       ; SALVA CODICE CONT.
C407- 20 D9 C0  8300            JSR CTRLBLOCK       ; INIZIALIZZA SCHEDA!
C40A- B0 18     8310            BCS ERREXIT  ; ESCE SE ERR.
C40C- A5 00     8320            LDA *R/WFLAG        ; LETTURA O SCRITTURA ?
C40E- F0 03     8330            BEQ DISKW    ; SCRITTURA!
C410- A9 00     8340            LDA #READCOMP       ; CODICE CONTR. LETTURA
C412- 2C        8350            .BY $2C
C413- A9 01     8360 DISKW      LDA #WRITCOMP       ; CODICE CONTR. SCRITT.
C415- 85 02     8370            STA *SAVEPARM       ; SALVA PAR. CONTR.
C417- 20 D9 C0  8380            JSR CTRLBLOCK
C41A- B0 08     8390            BCS ERREXIT
C41C- A9 20     8400            LDA #DEINIT  ; DEINIZIALIZZA
C41E- 85 02     8410            STA *SAVEPARM       ; SALVA COD. CONTR.
C420- 20 D9 C0  8420            JSR CTRLBLOCK       ; ESEGUE DEINIZ.
C423- 60        8430 GOODRET    RTS  ; USCITA OK!
                8440            ;
C424- 20 09 C0  8450 ERREXIT    JSR DRIVEOFF        ; PREPARA PER SPEGNERE
C427- 20 61 C0  8460            JSR MOTOR    ; I MOTORI COMUNQUE
C42A- 60        8470            RTS  ; TORNA (ERRORE IN ERRCODE ()$FF)
                8480            ;
                8490            ;
                8500            .EN

LABEL FILE:  [ / = EXTERNAL ]

R/WFLAG=0000        NDISK=0001          SAVEPARM=0002
NSCHEDA=0003        FDCCOM=0004         ERRCODE=0005
NDRIVER=0006        FDCSTATUS=0007      NTRACK=0008
NSECTOR=0009        DRSTATUS=000A       INITADDR=000B
ENDADDR=000D        NTENT=000F          NERR=0010
/CONF=000A          BASE=0011           START=C000
/VIA=A880           /DRB=A880           /DRA=A881
/DDRB=A882          /DDRA=A883          /T1CL=A884
/T1CH=A885          /T1LL=A886          /T1LH=A887
/T2CL=A888          /T2CH=A889          /ACR=A88B
/PCR=A88C           /IFR=A88D           /IER=A88E
/INIT=0010          /INIT1=0030         /DEINIT=0020
/READCOMP=0000      /WRITCOMP=0001      /READ=0003
/WRITE=0004         /COMPARE=0006       SETDRIVE1=C003
SETDRIVE2=C006      DRIVEOFF=C009       VIAINIT=C00E
FORCEINT=C02B       PUNTSCHED=C04B      FORCE=C053
MOTOR=C061          SETWRITE=C06B       SETREAD=C078
INDCSR=C085         INDSR=C08D          INDDR=C097
EXESTA=C09C         EXECCOM=C0A0        DSYSIRQ=C0B4
IRQDIS=C0C1         READSTATUS=C0C8     CTRLBLOCK=C0D9
INIT1ENT=C0E7       INIT1RET=C0F5       INITENT=C0F8
INITRET=C106        DEINITENT=C109      DEINITRET=C11F
DIVERSO=C122        LOOP1=C132          LOOP7=C13F
LOOP6=C144          READENT=C165        WRITENT=C173
STATERR=C17D        BRLOOP1=C180        BLKRET=C182
BLKRET1=C18B        READ/CENT=C18C      WRIT/CENT=C18F
OKRET=C1A2          COMPENT=C1AF        CSTATERR=C1C6
LOOP2=C1D1          LOOP3=C1F0          LOOP4=C1F6
LOOP5=C201          DRIVCTRL=C20A       DCTR1=C216
DCTR4=C231          DCTR2=C23C          DCTRET=C23F
DCTR3=C240          DISCVER=C252        DVER2=C267
DVER1=C277          DCTRET1=C287        DVERRET=C288
TRINDR=C289         INDTR=C2A0          SEEK=C2AB
EXECCOM1=C2BA       SEEKSTOP=C2BA       SEEK1=C2BF
SEEK2=C2D5          SEEKERR=C2DE        NSECINSR=C2E0
VERIFY=C2EF         VER1=C2FF           SETADDR=C305
STAD1=C315          STADRET=C319        READSECT=C31A
RSEC1=C327          RSEC2=C338          RSEC3=C347
RSEC5=C350          WRITSECT=C35B       WSEC1=C365
WSEC2=C376          COMPSECT=C386       CSEC0=C393
CSEC1=C39A          CSEC2=C3A6          CSEC3=C3B5
RETRY=C3BA          RETRY1=C3CC         RETRY2=C3CE
RETRY3=C3DF         RETRY4=C3EE         RETRY5=C3F3
START1=C3F9         DISK1=C403          DISKW=C413
GOODRET=C423        ERREXIT=C424
//0000,C42B,C42B
)
```

Per ragioni di spazio, il proseguimento del listato verrà presentato nella prossima puntata.

lume Table Of Contents", ossia la famigerata VTOC (che troviamo soprattutto sui grossi sistemoni tipo IBM e simili). La VTOC contiene una pedissequa descrizione di come sono allocati gli "n" settori delle "m" tracce del disco. Orbene si supponga che DOS riceva il primo comando di *SAVE*, per un file che attualmente in RAM risiede dall'indirizzo XX fino all'indirizzo YY.
La prima azione del DOS è una lettura della VTOC, da cui deduce che *tutti* i settori sono liberi. Ha inizio allora il riempimento - con i dati del file - dei settori della traccia più vicina (es.: 19). Se questa non basta, si passa ad un'altra traccia contigua. Il lavoro termina così: si scrive la lista, si aggiorna l'indice (o directori, che dir si voglia) e si aggiorna pure la VTOC. La procedura si ripete per successivi file, iniziando sempre con l'esame della VTOC. E' evidente che, in questa prima fase, i file occuperanno tutti i settori contigui, su tracce contigue. Ad un certo punto la VTOC è satura (o, perlomeno, i settori residui non contengono il nuovo file) così l'utente deve eliminare qualcosa di vecchio per poter scrivere il nuovo file. Dovrà quindi dire al DOS con un comando del tipo: "cancella il file di nome taldeitali". Tale ordine verrà eseguito semplicemente modificando la VTOC e l'indice, mentre l'effettivo contenuto dei settori impegnati in realtà rimane immutato (*ma ciò non importa: per tali dati si ha una specie di "morte civile" anche in attesa di venir ricoperti da dati nuovi.* NdR).
Ora si salva il nuovo file: possiamo supporlo più corto del precedente (altrimenti verrebbe richiesta l'uccisione di un altro file, mentre il caso di lunghezze identiche è poco probabile).
Sulla VTOC restano liberi dei settori fra uno dei vecchi file e quello nuovo. A furia di "deletare" / salvare file dopo un po' è facile comprendere che i settori liberi (e quindi utilizzabili) si mescolano in modo via via più complicato a quelli occupati. Se non vi fosse la lista già detta e se i file fossero brutalmente memorizzati uno dietro l'altro lo spreco di spazio sul disco diverrebbe ben presto tragico.
Anche senza voler qui imitare l'umorismo di un Paolo Villaggio, c'è da aggiungere qualche altra facezia analoga: si pensi che tra gli altri compiti del DOS c'è quello di avvisare l'utente su quanto è accaduto, qualora (si verifichino) condizioni di errore o qualora l'utente abbia proposto comandi ineseguibili, come secondo quanto già accennato sopra - quello di salvare un file mentre sul disco non resta spazio sufficiente). Il dialogo sarà da progettare in modo che l'utente possa sempre intervenire (ad esempio, il DOS avvisa: "la situazione e questa; che faccio? Proseguo o no?"; oppure: "Devi usare un altro nome per il file!"; "Hai chiesto di eliminare un file ma esso è protetto contro simili azioni: devo ... sproteggerlo?"). È buona norma che la sintassi dei comandi sia semplice e immediata. Ancora, vi devono essere protezioni contro azioni inconsulte o che potrebbero, per una svista, distruggere i contenuti del disco. E via di questo passo.
È dunque evidente che un DOS non si limita alla gestione di un trasferimento dei dati da/su disco, ma deve possedere un interprete dei comandi, una ricca biblioteca di messaggi facilmente comprensibili, e deve saper riconoscere le possibili situazioni di errore ed evidenziarle: tutto questo richiede un software assai "smaliziato" ed i possessori di un Apple (le idee sull'organizzazione del disco sono tratte dal relativo DOS) riconosceranno che un tale lavoro non deve essere stato facile nemmeno per i mitici Jobs e Wozniak! Per cui, rifacendomi a quanto detto nella premessa, invito a ritornare alla nostra umile e grezza RWTS (ancorchè funzionante benissimo) ed a considerare benevolmente l'idea di un DOS spartano ma utile per acquisire una notevole esperienza sui dischi. Così è stato anche per me, e se ora stiamo considerando l'idea di realizzare un controllore evoluto e con un DOS "lussuoso", il merito va in gran parte al 1771, alla semplice e robusta scheda di interfaccia, ed al DOS "3.06", col quale SYM ed AIM65 acquisiscono prestazioni decisamente buone, soprattutto in relazione ai bassi costi dell'unità a dischi e del software.
Le illustrazioni che corredano questa parte del discorso sulle unità a disco, riguardano: un elenco dettagliato dei comandi con i rispettivi flag ed i loro significati peculiari per il chip floppy disk controller (tavola 1); una somma delle capacità di memorizzazione su floppy disk per 5"1/4 e per 8", in singola e doppia densità (tavola 2); un esempio di applicazione dei nuovi chip di interfaccia (figura 1). Soprattutto in quest'ultima immagine, il Lettore potrà osservare come, con soli due chip, si costruisca il separatore di dati (valido per singola e doppia densità). Per il resto, adattando l'hardware della figura 1 allo schema della volta scorsa (tranne il data separator), si otterrà un controllore evoluto senza eccezionali modifiche (in pratica anche i chip FDC sono compatibili pin-to-pin). Ma, con l'arrivo degli FDC tutti-su-un-solo-chip, anche questa soluzione sarà superata, probabilmente già entro l'83.
Al termine di questa discussione due parole sui listati dei tre blocchi fondamentali del cuore di un DOS: una RWTS piuttosto a prova di siluro con un unico ritorno ed error-check incluso su tutte le combinazioni possibili di comandi; una "format" utile per generare un dischetto su cui sia poi possibile effettuare transazioni, ed una "recovery" che usa il comando di "read track". Il listato è sufficientemente commentato e, per una migliore comprensione, si consiglia di tenere sott'occhio la descrizione della parte hardware e gli schemi elettrici relativi. I drive consigliati sono: Basf, Shugart, Tandon, da 5"1/4 con singola testina di lettura/scrittura.
Ovviamente la scheda di interfaccia è disponibile, come pure la documentazione relativa, mentre, per la bibliografia, si rimanda alle note conclusive della seconda puntata.
(Per ragioni di spazio il proseguimento del listato verrà presentato nella prossima puntata).

(continua)

Per ogni ulteriori informazioni e contatti d'ogni genere:
Computerjob Elettronica,
Via A. Molinari n. 20
25100 Brescia
Tel. (030) 54878

una proposta interessante

real time associated

SISTEMA ABM.

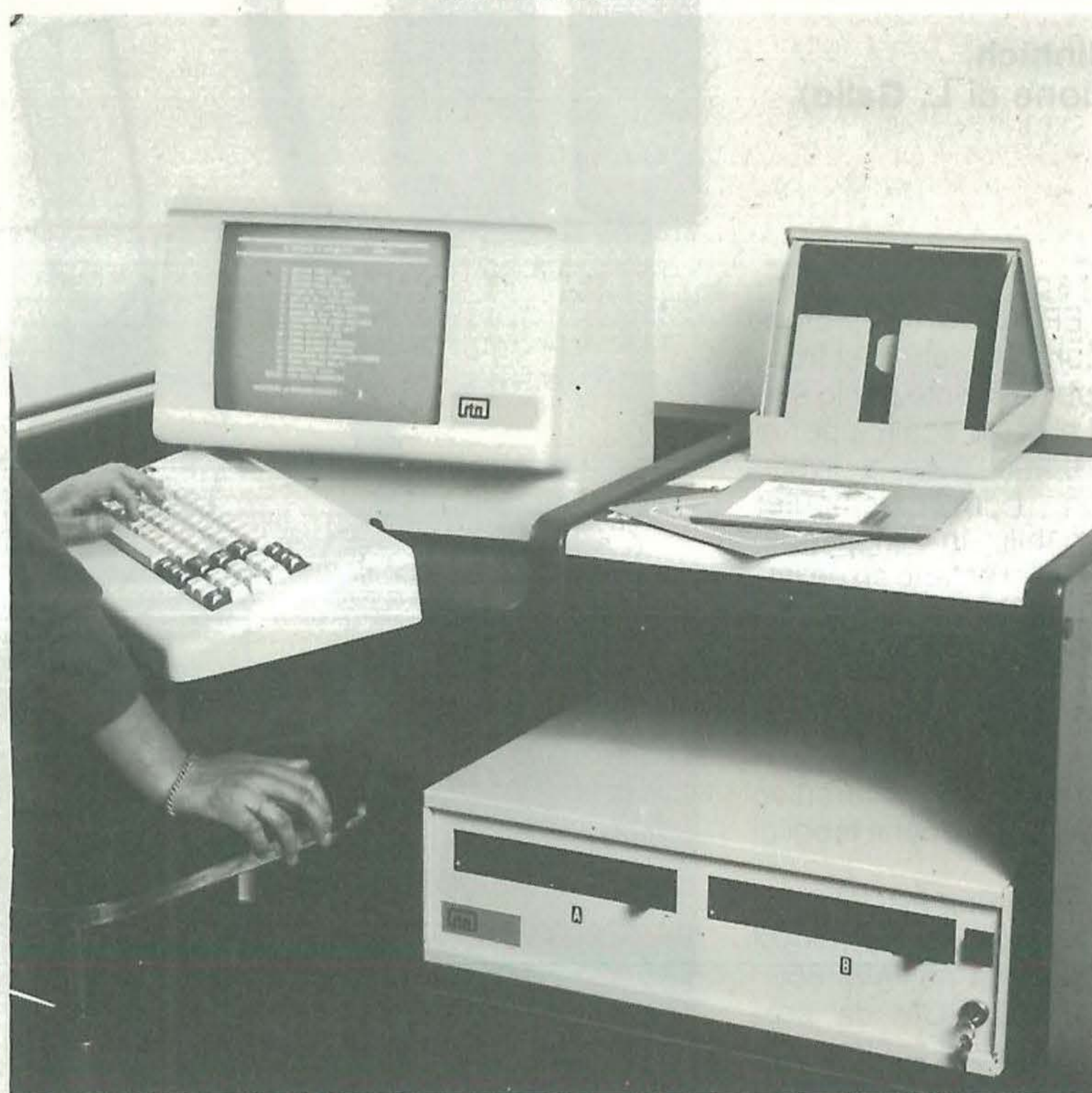
La RTA, Real Time Associated propone ai rivenditori una collaborazione globale e cioè:

Hardware
- —Monoscheda Z80 - 64K - CP/M
- —Video CRT 80x24 con tastiera alfanumerica separata
- —N° 2 unità floppy disk da 8" (2x500.000 car) IBM compatibili
- —Espansioni con hard disk da 5 o 10 Mbytes
- —Altissima affidabilità
- —Costo molto contenuto

Software
- —Oltre 500 programmi applicativi e gestionali collaudati e funzionanti
- —Oltre 800 installazioni in Italia e all'estero

Servizio
- —Manutenzione hardware e software

— Produzione di software di base e applicativo — Corsi di vendita beni strumentali — Corsi per tecnici hardware — Corsi su software applicativo, di base e linguaggi — Consulenza marketing, promozioni e pubblicità. Tutto questo è quanto la RTA offre ai suoi rivenditori. Professionalità e know-how per un servizio globale.

Parliamone insieme: questa è una proposta interessante.

rta real time associated
viale Gran Sasso, 3 - 20131 MILANO
telefoni 02/2040614 - 2046371

Per saperne di più; compilate questo tagliando e speditelo in busta chiusa alla RTA

NOME ______

COGNOME ______

SOCIETÀ ______

INDIRIZZO ______ TEL. ______

C.A.P. ______ CITTÀ ______

Chiedere del Signor ______

di **H. Minnich**
(traduzione di **L. Gallo**)

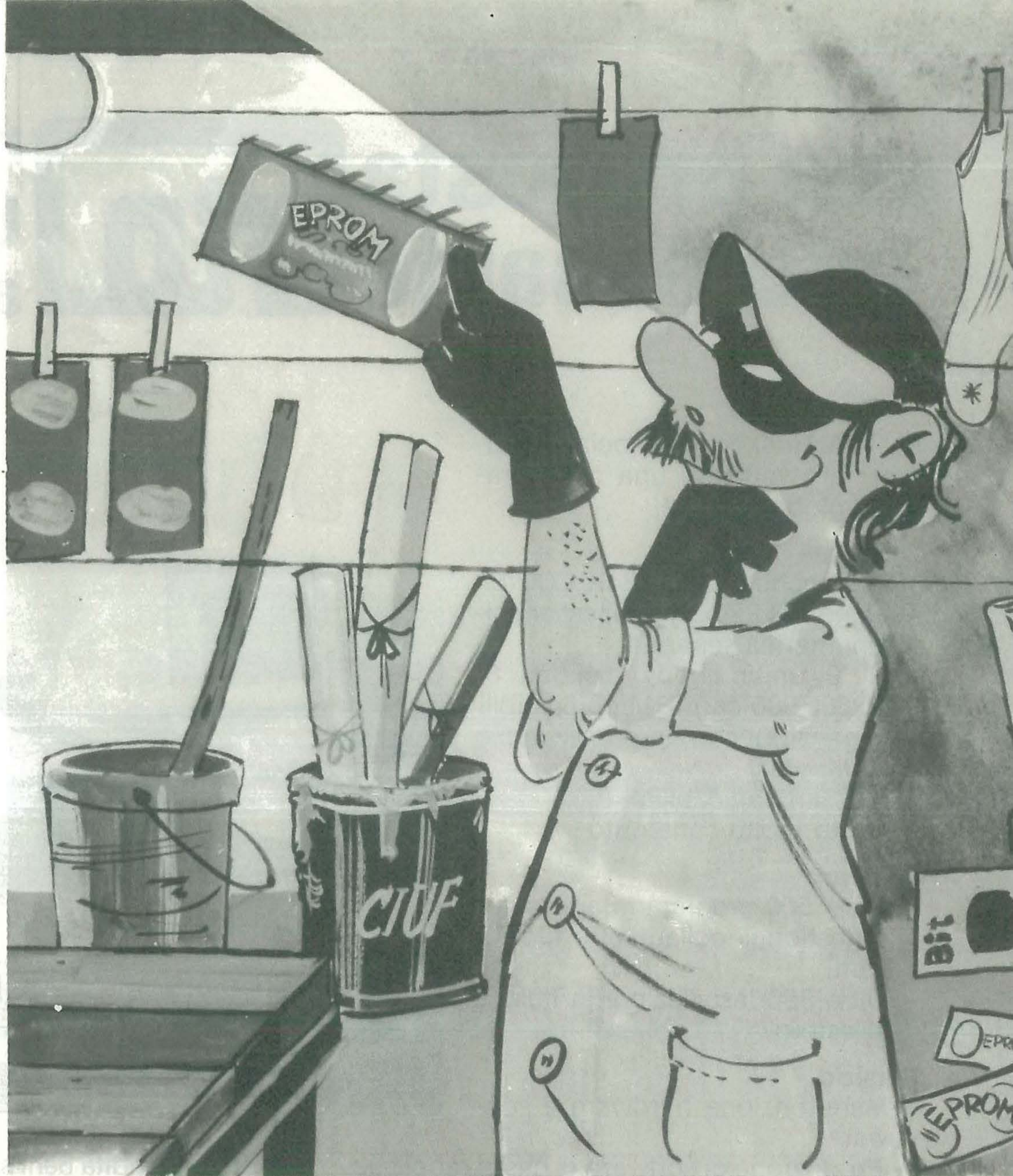
I costi per apparecchi programmatori di EPROM di tipo professionale sono in molti casi troppo alti, specialmente per lo sviluppo su piccola scala e per la ricerca, come pure a livello hobbistico. I microcomputer programmabili in linguaggio macchina si prestano ad un impiego alternativo, a condizione che posseggano un minimo di possibilità di dialogo.
Sfruttando la routine *PRTSTR* /2/ è stato sviluppato un comodo programma interattivo che gestisce i seguenti modi di funzionamento:

- copiatura di una PROM da 1, 2 o 4 Kbyte in un'area di RAM a libera scelta nell'AIM 65;
- test delle EPROM da programmare, per controllarne la completa cancellazione;
- trasmissione dei dati situati in aree di RAM a scelta in una EPROM da 1, 2 o 4 Kbyte;
- verifica dei dati trasmessi alla EPROM, per controllarne l'identità con quelli contenuti nell'area RAM d'origine.

L'hardware

Si è adottata una soluzione che può funzionare già con la versione base dell'AIM 65, nella quale l'utente dispone di due porte d'ingresso/uscita del VIA 6522. Una porta viene utilizzata per il traffico bilaterale dei dati fra l'AIM 65 e le linee-dati della piastra di programmazione di EPROM.
La seconda porta di ingresso/uscita serve per pilotare tre spie di stato sulla piastra stessa e per sorvegliare lo stato di inserimento della tensione continua di 25 V necessaria alla programmazione.
Non restano così già a priori abbastanza linee per gestire l'indirizzamento della EPROM tramite le porte d'utente. Una soluzione alternativa adatta allo scopo consiste nell'utilizzo di un contatore esterno da 12 bit, al quale vengono forniti in forma appropriata gli impulsi di reset e di conto tramite la porta d'utente: tramite la linea di controllo di stato CA2 viene portato a livello alto il pin di programmazione della EPROM, CB2 fornisce gli impulsi di conteggio e, attraverso un'ulteriore linea della porta utilizzata come uscita, viene emesso il segnale di chip-select.
L'hardware necessario consta così di una piastra da collegarsi con la spina per applicazioni. Benchè il software tenga conto delle commutazioni necessarie a seconda del tipo di EPROM da programmare, occorre collegare l'uscita PB2, tramite la quale vengono applicati i 24 V alla EPROM, nel caso della programmazione di una TMS 2532 (quando si debbono programmare EPROM da 1 a 2 Kbyte, PB2 si trova costantemente alto).
L'indirizzo A11 viene fornito mediante PB1 per tenere automaticamente in considerazione i differenti pin-out.

Il software

Nella realizzazione del modulo per la programmazione di EPROM ci si è attenuti scrupolosamente ai timing specificati dalle ditte produttrici di EPROM. Ogni EPROM da programmare viene controllata prima dell'effettivo processo di programmazione, per vedere se è completamente cancellata, onde escludere così, in seguito, ogni dubbio durante il confronto passo-passo dei dati scritti con quelli originali che si trovano nella RAM. Qualora venga scoperto un indirizzo di EPROM non libero compare un avviso d'errore ed il LED rosso di spia sulla piastra di programmazione si mette a lampeggiare.
Dopo aver stabilito la grandezza della EPROM (domanda "*SIZE*") ed aver introdotto l'indirizzo iniziale dell'area di RAM da trasmettere, viene controllata la presenza della tensione di programmazione di 25 V ed i dati vengono trasmessi dalla RAM alla EPROM.
Dopo ogni passo vengono letti di nuovo dalla EPROM e confrontati con i dati originali. Se i dati corrispondono, viene incrementato il contatore d'indirizzo.
Nel caso che i dati non siano

L'AIM programma le EPROM

PRESENTAZIONE

Il microcomputer AIM 65 e il Siemens PC 100 offrono posto per una memoria RAM da 4 Kbyte già sulla loro piastra base. Questo fatto viene sfruttato nel modulo di software descritto nel presente articolo, che rende questi apparecchi confortevoli programmatori di EPROM con la sola tensione di alimentazione di 5 V.
Il programma tiene del tutto conto del differente pin-out dei tipi 2758, TMS 2516 o MK 2716 da una parte e del TMS 2532 dall'altra, senza peraltro rendere necessaria una commutazione nell'hardware.

Figura 1 - Hardware per programmare tutti i tipi comuni di EPROM con i microcomputer AIM 65 e PC 100.

identici si emette un avviso d'errore e si comunica l'indirizzo in corrispondenza del quale l'errore ha avuto luogo.
L'utente ha anche la scelta, per mezzo di un interruttore software, di trasferire il contenuto di una EPROM programmata, in un'area di RAM a scelta.
In questo caso, alla fine del trasferimento dell'ultimo byte, viene segnalata la conclusione del processo e posta la domanda se si desidera passare alla fase di programmazione di EPROM o se il programma debba terminare, per passare eventualmente alla fase di elaborazione dei dati trasmessi alla memoria RAM.

Uso del programma

Dopo aver collegato la piastra per programmazione di EPROM con la spina per applicazioni e con una tensione continua di 25 V si può far partire il programma, già caricato nella RAM. L'avvio del programma si effettua inserendo l'indirizzo di start:

★ = 0C00
(G)/SPACE

oppure con un salto a tale indirizzo da un altro programma d'utente. Se l'interruttore per i 25 V si trova in posizione "on", compare l'avviso:

TURN OFF 25 V!

Dopo aver girato l'interruttore si spegne la spia gialla L3 ed il programma procede, inizializza alcuni registri e pone il quesito:

COPY(C),BURN(B):

A questa domanda si deve rispondere digitando *C* o *B*. Qualsiasi altra risposta non è accettata dal programma, che pone nuovamente la domanda. La risposta *C* consente di copiare i dati immagazzinati in una PROM nell'area RAM libera dell'AIM 65. In tal caso viene richiesto l'indirizzo iniziale dell'area RAM da occupare:

COPY FROM ...

La risposta deve corrispondere all'indirizzo iniziale in forma esadecimale a quattro cifre, facendo molta attenzione che il programma di controllo non sia distrutto dalla copia della PROM. In seguito si ha la richiesta della grandezza della PROM da trasmettere (in Kbyte):

EPROM SIZE = 1, 2, 4 K?

Risposte diverse da 1, 2 o 4 sono rifiutate dal programma, che ripete la domanda. Se la risposta è corretta, viene copiato il contenuto della PROM nell'area RAM prescelta. Se il programma tenta di scrivere in un'area non contenente memorie RAM, compare l'avviso:

MEMFAIL XXXX

dove *XXXX* sta per l'indirizzo della prima cella di memoria non scrivibile, quindi si verifica il rientro al monitor. Se la trasmissione nell'area RAM si è conclusa con successo appare la scritta:

TRANSFER OK, GO ON?

Rispondendo con *Y*, si salta di-

L'AIM programma le EPROM

rettamente alla fase "burn", altrimenti si abbandona *(N)* correttamente il programma tornando al monitor, dove si offre la possibilità, ad esempio, di modificare i dati della RAM ecc. Ogni altra risposta alla domanda *GO ON?* è scartata e la domanda riformulata.
Sulla risposta *Y* si giunge al medesimo punto del programma al quale si sarebbe pervenuti rispondendo *B* alla domanda *COPY(C),BURN(B)*. Il programma chiede adesso la capacità della EPROM da programmare con:

PROMSIZE = 1, 2, 4 K?

al che il numero corrispondente deve essere comunicato al calcolatore, il quale controlla che l'EPROM sia interamente cancellata. Se una cella della EPROM risulta non vuota, il computer lo segnala col messaggio:

EPROM NOT EMPTY .

e, inoltre, la spia rossa sulla piastra di programmazione si mette a lampeggiare. A questo punto si può abbandonare il programma premendo il tasto *ESC*. Qualora invece l'esame della EPROM abbia avuto successo, compare l'invito ad inserire la tensione continua di 25 V:

TURN ON 25 V.

Solo dopo che i 25 V sono stati connessi compare la domanda relativa all'area RAM da immagazzinare nella EPROM:

RAM STARTADR. = ...

Nella risposta si deve precisare l'indirizzo iniziale, in forma esadecimale. In seguito i dati vengono scritti passo-passo dalla RAM nella EPROM. Dopo ogni ciclo di scrittura si passa alla lettura dell'informazione appena scritta e al confronto con quella originale della RAM. Se i due contenuti non collimano compare la scritta:

EPROM ERROR IN YYYY

dove *YYYY* sta per l'indirizzo dal quale è stata letta l'informazione errata. Contemporaneamente si accende il LED rosso. Normalmente compaiono sul display tre blocchi che variano in continuazione:

XXXX YYYY ZZ

qui *XXXX* è l'indirizzo corrente della RAM da cui vengono presi i dati, *YYYY* è l'indirizzo attuale della EPROM, che viene sempre incrementato a partire da zero, e *ZZ* sono i dati.
Lo svolgimento automatico del programma può essere arrestato senza pericolo premendo il tasto *SPACE* e, in tutti i casi, continuato passo-passo. Dopo aver scritto e controllato il contenuto dell'ultimo indirizzo, dato dalla capacità della EPROM, il programma si fa vivo per l'ultima volta con la scritta

EPROM OK! 25 V OFF!

e subito dopo accende il LED di controllo verde. Girando l'interruttore dei 25 V si trasferisce il controllo al monitor.

0C00	A9	F5	8D	02	A0	AD	00	A0	29	08	F0	08	A9	40	8D	00
0C10	A0	4C	E3	0E	A9	00	8D	00	A0	8D	03	A0	A9	CC	8D	0C
0C20	A0	20	44	EB	20	C0	0E	43	4F	50	59	28	43	29	2C	42
0C30	55	52	4E	28	42	29	3A	EA	20	3C	E9	C9	43	F0	07	C9
0C40	42	D0	DE	4C	DF	0C	20	44	EB	20	C0	0E	43	4F	50	59
0C50	20	46	52	4F	4D	20	20	20	EA	20	1D	0F	20	44	EB	20
0C60	C0	0E	50	52	4F	4D	20	53	49	5A	45	20	3D	31	2C	32
0C70	2C	34	4B	20	3F	EA	20	33	0F	D0	E1	20	60	0F	A9	8D
0C80	8D	27	A4	A9	CD	8D	2A	A4	A9	60	8D	2D	A4	A6	FF	10
0C90	03	20	A7	0F	18	98	65	F6	85	FD	D0	02	E6	FE	AD	01
0CA0	A0	20	D3	0F	F0	03	4C	BE	0F	20	28	0D	F0	03	4C	8D
0CB0	0C	20	44	EB	20	C0	0E	54	52	41	4E	53	46	45	52	20
0CC0	4F	4B	3B	20	47	4F	20	4F	4E	3F	EA	20	73	E9	C9	59
0CD0	D0	03	4C	DF	0C	C9	4E	D0	D8	20	F0	E9	4C	A1	E1	20
0CE0	44	EB	20	C0	0E	45	50	52	4F	4D	20	53	49	5A	45	3D
0CF0	31	2C	32	2C	34	4B	20	3F	EA	20	33	0F	D0	E1	A9	CC
0D00	8D	0C	A0	A6	FF	10	08	A9	01	0D	00	A0	8D	00	A0	20
0D10	60	0F	A5	FF	10	03	20	A7	0F	AD	01	A0	49	FF	D0	25
0D20	20	28	0D	F0	3A	4C	12	0D	AD	0C	A0	09	20	8D	0C	A0
0D30	29	DF	8D	0C	A0	E6	FB	D0	02	E6	FC	C8	D0	02	E6	F7
0D40	A5	FC	C5	F8	60	20	44	EB	20	C0	0E	45	50	52	4F	4D
0D50	20	4E	4F	54	20	45	4D	50	54	59	20	EA	4C	00	0F	20
0D60	44	EB	20	C0	0E	54	55	52	4E	20	4F	4E	20	32	35	20
0D70	56	20	20	20	20	20	20	20	EA	AD	00	A0	29	08	F0	F9
0D80	20	A0	0E	AD	00	A0	29	08	F0	EF	A6	FF	30	02	09	44
0D90	09	40	8D	00	A0	20	44	EB	20	C0	0E	52	41	4D	20	53
0DA0	54	41	52	54	41	44	44	52	2E	3D	EA	20	1D	0F	20	60
0DB0	0F	A6	FF	10	03	20	A7	0F	AD	00	A0	09	01	8D	00	A0
0DC0	A6	FF	10	08	A9	04	0D	00	A0	8D	00	A0	A9	FF	8D	03
0DD0	A0	E6	FD	D0	02	E6	FE	20	44	EB	A5	FE	20	46	EA	A5
0DE0	FD	20	46	EA	20	3B	E8	A5	FC	20	46	EA	A5	FB	20	46
0DF0	FA	20	3B	E8	20	3B	F8	B1	F6	48	20	9E	EB	20	46	EA
0E00	20	07	E9	20	AC	EB	68	8D	01	A0	A6	FF	10	03	4C	87
0E10	0F	AD	0C	A0	09	02	8D	0C	A0	20	B3	0E	AD	0C	A0	29
0E20	FD	8D	0C	A0	AD	00	A0	29	FE	8D	00	A0	A9	00	8D	03
0E30	A0	AD	01	A0	D1	F6	D0	08	20	28	0D	F0	31	4C	B1	0D
0E40	AD	00	A0	09	10	8D	00	A0	20	44	EB	20	C0	0E	45	50
0E50	52	4F	4D	20	45	52	52	4F	52	20	49	4E	20	EA	A5	FC
0E60	20	46	EA	A5	FB	20	46	EA	20	07	E9	4C	68	0E	20	44
0E70	EB	20	C0	0E	45	50	52	4F	4D	20	4F	4B	21	20	32	35
0E80	56	20	4F	46	46	21	EA	AD	00	A0	09	20	8D	00	A0	20
0E90	75	0F	AD	00	A0	29	9F	8D	00	A0	20	F0	E9	4C	A1	E1
0EA0	A9	20	8D	04	A0	A9	4E	8D	05	A0	2C	0D	A0	50	FB	AD
0EB0	04	A0	60	A9	50	8D	04	A0	A9	C3	8D	05	A0	4C	AA	0E
0EC0	68	85	F9	68	85	FA	20	9E	EB	E6	F9	D0	02	E6	FA	A0
0ED0	00	B1	F9	C9	EA	F0	09	20	7A	E9	18	90	EC	20	AC	EB
0EE0	6C	F9	00	20	44	EB	20	C0	0E	54	55	52	4E	20	4F	46
0EF0	46	20	32	35	20	56	20	21	20	EA	20	75	0F	4C	14	0C
0F00	A9	FF	8D	04	A0	A9	FF	8D	05	A0	2C	0D	A0	50	FB	AD
0F10	00	A0	49	10	8D	00	A0	20	07	E9	4C	05	0F	20	FD	E3
0F20	85	F7	85	FE	20	FD	E3	85	F6	38	E9	01	B0	02	C6	FE
0F30	85	FD	60	20	73	E9	C9	31	F0	09	C9	32	F0	0A	C9	34
0F40	F0	0B	60	A9	04	4C	4F	0F	A9	08	4C	4F	0F	A9	10	85
0F50	F8	E9	10	F0	04	A9	00	F0	02	A9	80	85	FF	A9	00	60
0F60	AD	00	A0	09	80	8D	00	A0	29	7F	8D	00	A0	A9	00	85
0E70	FB	85	FC	A8	60	AD	00	A0	29	08	D0	F9	20	A0	0E	AD
0F80	00	A0	29	08	D0	EF	60	AD	00	A0	29	FE	8D	00	A0	20
0F90	B3	0E	AD	00	A0	09	01	8D	00	A0	29	FB	8D	00	A0	29
0FA0	FE	8D	00	A0	4C	2C	0E	AD	00	A0	29	02	D0	07	A9	FD
0FB0	2D	0C	A0	D0	05	A9	02	0D	0C	A0	8D	0C	A0	60	18	98
0FC0	65	F6	8D	1C	A4	90	02	E6	F7	A5	F7	8D	1D	A4	A0	00
0FD0	4C	33	EB	48	A5	FD	8D	28	A4	8D	2B	A4	A5	FE	8D	29
0FE0	A4	8D	2C	A4	68	4C	27	A4	78	5A	55	D9	4E	5B	23	A3

Figura 2 - Il software del "programmer" è lungo 1 Kbyte ed offre un confortevole funzionamento interattivo.

Bibliografia

- Doris, A.: Programmateur d'EPROM. Micro Systems X/79.
- Franke, E.: PRTSTR, Funkschau-Sonderheft Hobbycomputer II.
- AIM 65 User Manual ■

Non contenti, insistono

Tutte le volte che capita, cerco fermamente di trattenermi, di lasciar perdere. Ma è più forte di me, non riesco a farne a meno, è incontrollabile.
Ma cosa pensate? Non sto parlando di un vizio indecoroso, ma dell'incontenibile rabbia che mi assale tutte le volte che leggo articoli, speciali, note, redatte da giornalisti inesperti ed approssimativi, ed allegramente pubblicati, con addirittura dedica di copertina, dalla stampa settimanale a larga diffusione.
"L'uomo è nervoso, irritabile", forse dirà di me qualcuno, parafrasando definizioni con cui vengono rappresentati i nostri politici, soprattutto se carismatici.
Non è affatto vero. Non sono nè nervoso nè irritabile.
Solo non sopporto il ridicolo, la disinformazione spacciata per novità e cultura alle masse.
In ogni caso, l'avrete già capito, mi riferisco all'articolo apparso sull'Espresso del 23/1/83 pag. 42 e seguenti.
Mi spiace solo che questa **Nota** *per programmazione interna alla Redazione di* **Bit**, *uscirà troppo lontano dalla data citata, per permettere un immediato collegamento. Comunque sia, andando a pag. 43, si legge "Anche in Italia, nasce l'uomo computer".*
Anche....., cioè? Nel senso di: finalmente, oggi, da adesso in poi, malgrado tutto, per quanto gli italiani....?
Oppure: dopo USA, Giappone, questo, quello, ecco che anche in Italia, ultimo fanalino disgraziato...?
Siamo nel 1983 e negli anni passati ne sono successe di cose, "malgrado" i notisti di cultura dell'Espresso.
Poi ancora:
"Nella scuola si fanno esperimenti sporadici..."
Sporadici? Ma scherziamo? Signori, leggete, documentatevi, smettete di guardare le modelle delle vostre foto che amoreggiano con l'Apple o con l'M20, quasi si trattasse di deodoranti o profumi.
Vi dirò: a me "piace" leggere questo tipo di articoli, ne ricavo un gusto perverso, il gusto dell'orrido.
Sentite questa: "Certo, non raggiungeremo mai il livello degli Stati Uniti (dove il personal computer è stato ideato nel 1975 da due tecnici californiani, Steve Jobs e Steve Wozniak)". Ah! Preso in castagna. Orribile. Il PC è stato ideato nel 1975, e per di più da due persone. E gli altri?
Allora, XYZ ideò la ruota, Meucci ideò il telefono, ed i nostri della Apple idearono il PC. E perchè allora non lo brevettarono visto che l'avevano ideato, di sana pianta, così, dall'oggi al domani? Ma non è finita, andiamo oltre.
"Del resto, nella scuola ufficiale è silenzio" Che ne dite? Qualcuno vuol parlare? Aspettate.
".... in compenso, dalle Alpi allo Stretto germogliano in ordine sparso esperimenti spontanei, selvaggi. Le riviste specializzate di Elettronica ospitano spesso i resoconti entusiastici di improvvisati maestri super-bit". Accidenti! Questa sì che è completa conoscenza della situazione italiana: dalle Alpi allo Stretto. E voi super-bit improvvisati, che ne dite?
Ora, mi si potrebbe dire che è ben facile criticare, fare "le pulci" su una singola frase od espressione usata. Ma qui non si tratta di questo. Si tratta dell'intero testo che tende a divulgare presso le masse un'immagine distorta, falsa e deleteria dell'attuale stato dell'arte dei PC e del loro utilizzo in Italia.
Forse, l'unica persona che ne viene fuori bene è la Professoressa Margherita Fasano, definita "la massima teorica italiana dell'informatica povera (?)".
Cara Margherita, in effetti ti sei sempre battuta per questa causa, ma se hai parlato con l'estensore dell'articolo de l'Espresso, purtroppo penso che non sei riuscita più di tanto a fargli capire come stanno le cose. Ma non è colpa tua. La colpa è della diffusa arroganza giornalistica: io sono giornalista, e come tale posso senza problemi, pontificare su questo e quello, alla faccia di tutto.
Eh, no! No, Signor Tullio Fazzolari, Sig.ra o sig.na Cristina Mariotti.
In quanto autori dello "speciale" avreste dovuto usare più umiltà, più informazione, più cultura.
Non è così che si aiuta lo sviluppo tecnico-culturale di una nazione.

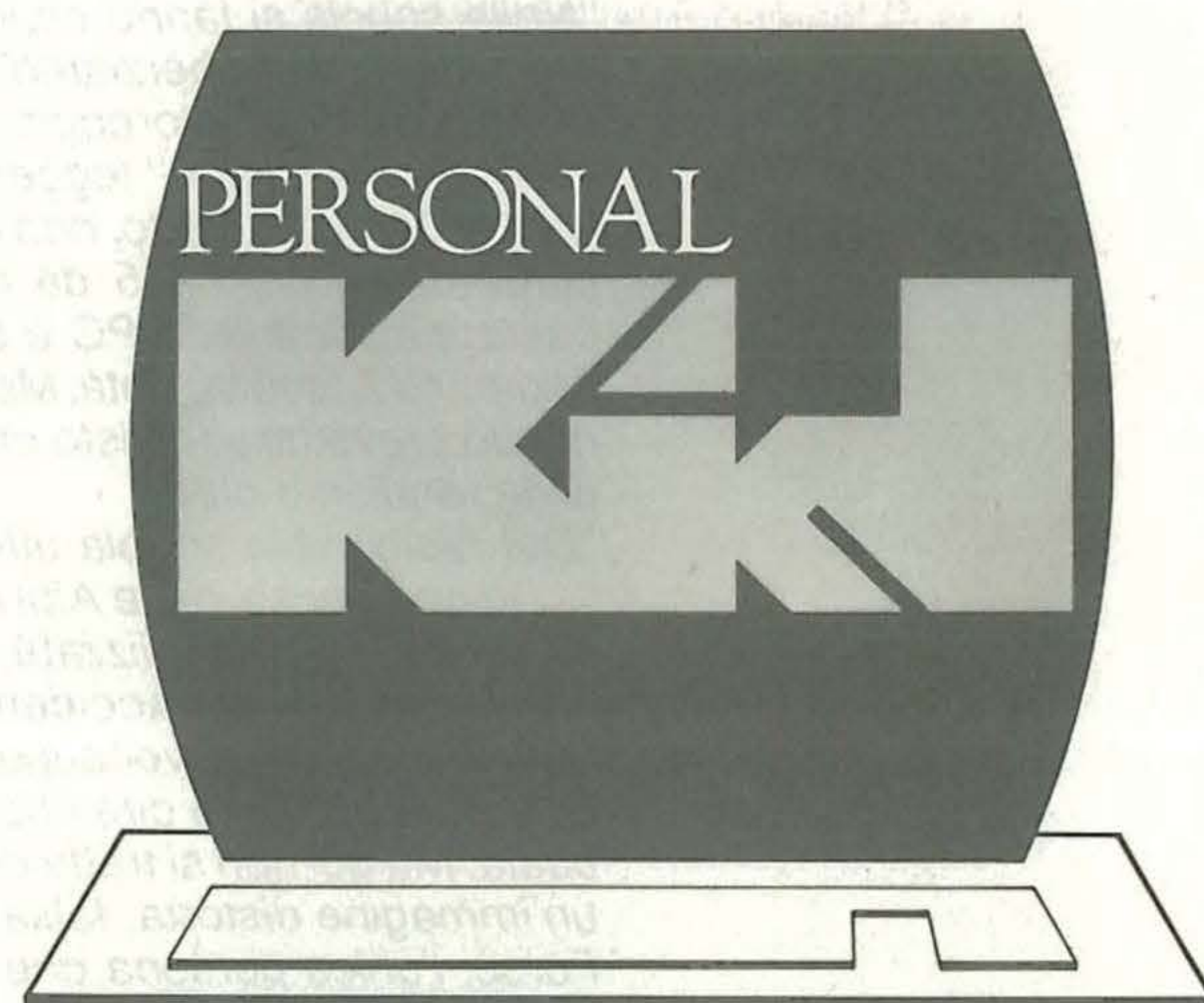
Aldo Cavalcoli

Che cosa ha in più Personal Kid?

	PREZZO (IVA escl.)
CPU BOARD 48 K RAM	650.000
Tastiera ASCII con pad numerico esteso e tasti funzionali	210.000
Alimentatore 80 W	150.000
Alimentatore switching 75 W	200.000
Contenitore	120.000
UNITÀ CENTRALE (48 K RAM, interfaccia per registratore, input analogici, lettere minuscole, BASIC, monitor e disassembler) completa di alimentatore, tastiera ASCII dotata di pad numerico esteso e tasti funzionali, contenitore	
Con tastiera incorporata	1.210.000
Con tastiera separata	1.260.000
UNITÀ CENTRALE con monitor	
Con tastiera incorporata	1.450.000
Con tastiera separata	1.500.000
UNITÀ CENTRALE con monitor 12", drive 5" e interfaccia per due drive	
Con tastiera incorporata	2.250.000
Con tastiera separata	2.300.000
Monitor 12" fosfori verdi o gialli	250.000
Drive 5"	710.000
Interfaccia doppio drive	120.000
Espansione 16 K RAM	150.000

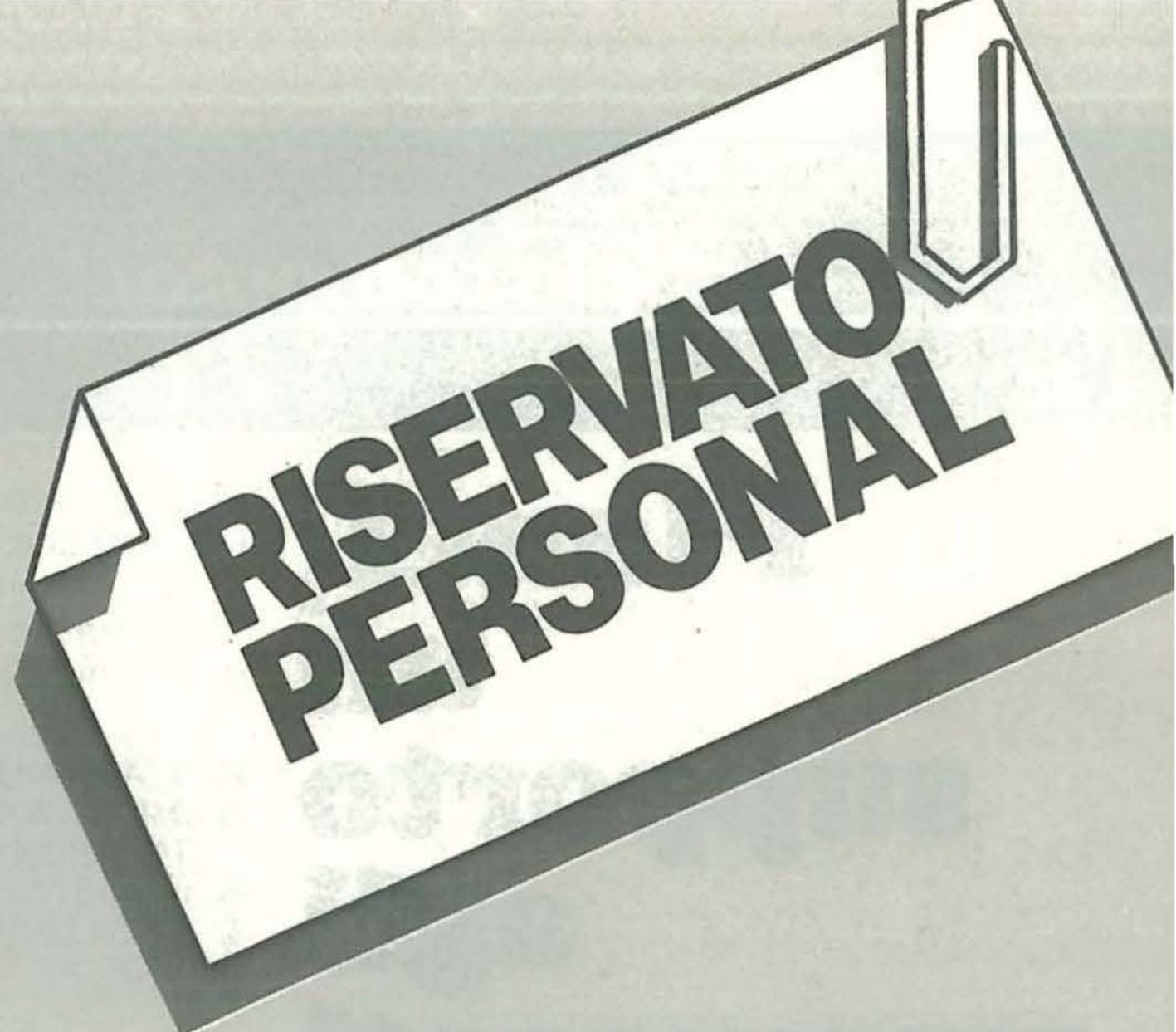
- Costo Basso
- Lettere minuscole
- Tastiera con pad numerico + i segni delle operazioni
- Repeat automatico
- Set di tasti funzionali per l'esecuzione immediata dei principali comandi
- Completo controllo del cursore
- Zoccolo per memoria EPROM
- Disponibilità del sistema in versione open frame o vestita in più configurazioni

Compatibile Apple

Marketing plan - ANCONA

SIPREL s.r.l. Via Di Vittorio, 82 - Zona Ind. Baraccola 60020 - Candia di Ancona
ANCONA TEL. 071/8046305 - MILANO TEL. 02/487930 - BOLOGNA TEL. 051/346013 -
PESCARA TEL. 085/378195 - PISA TEL. 050/575480

Cercasi Concessionari

Bit

RISERVATO PERSONAL

Planel, un supporto agli smemorati

Maxi-calendario con festività, compleanni, appuntamenti. Un programma di pianificazione del proprio prezioso tempo.

a cura della Redazione

Il *Planel* è un esauriente raccoglitore di dati per pianificare la vostra agenda, le feste, i giorni di appuntamento per mesi, e perfino anni nel futuro. Potete produrre un calendario per uno o più mesi, per il 1983, o il 1985, o il 1989 o come preferite.
Inoltre potete ottenere che il programma vi stampi le feste sul calendario, oppure voi stessi potete introdurre le vostre scadenze importanti: l'anniversario del matrimonio, il compleanno dei vostri genitori e così via.
Planel è un programma Applesoft che richiede almeno una stazione dischi e una stampante. Per lanciarlo è sufficiente digitare *RUN*.

Si presenterà la pagina con l'intestazione dalla quale premendo un qualsiasi tasto, verrà visualizzato il seguente menu:

1) STAMPA CALENDARIO
2) IMPOSTA/MODIFICA DATE IMPORTANTI
3) IMPOSTA GIORNI FESTIVI
4) LISTA DATE IMPORTANTI
5) FINE

da cui sceglierete l'opzione di vostra preferenza.

Stampa calendario

Questa opzione inizia la stampa di un calendario, prima che tale abbia inizio vengono poste quattro domande:

DAL MESE (GEN):
AL MESE (DIC):
ANNO (19XX):
OK? (S):

Alle richieste *DAL* e *AL MESE* si può rispondere sia con *RETURN*, accettando così l'opzione corrente in memoria segnata tra parentesi, sia digitando le tre lettere iniziali dei mesi necessari.
Se si vuole stampare da Settembre a Dicembre si può impostare *SET* e *RETURN* mentre per avere un solo mese si imposta la stessa sigla a entrambe le richieste. Lo stesso vale per gli anni: si può lasciare l'anno previsto in memoria (viene caricato il 1981) oppure stabilirne uno qualsiasi tra il 1981 e il 2000. L'anno già previsto è quello il cui file delle scadenze è caricato nella memoria dell'Apple. Impostandone uno diverso si riceve la notifica: *19XX IL FILE NON È IN MEMORIA* e quindi la richiesta *CARICO DA DISCO?*. Rispondendo *"S"* si carica da disco un file precedentemente creato per quell'anno, mentre con *"N"* si cancella il file in memoria e si prosegue senza alcun file delle date. L'ultima richiesta è l'*OK?*. Rispondendo con *S* o con *RETURN* si incomincia la stampa. A ciascuna di queste richieste si può abortire l'opzione e tornare al menu principale digitando una freccia verso l'alto (↑).
In figura 1 è riportato un campione del calendario che viene stampato. Si noti che l'anno e il mese appaiono in testa al foglio, il calendarietto in numeri, dei mesi precedente e seguente, appaiono in calce, e le festività e le scadenze importanti appaiono alla base della casella del giorno appropriato. I messaggi lunghi vengono suddivisi a metà, mandati a capo e mostrati su due righe. Se il messaggio impostato contiene uno slash (/) viene diviso in due messaggi nel punto dove c'è lo slash, e ciascun messaggio viene stampato su una riga preceduto da un punto.

Imposta-modifica date importanti

Questa opzione serve a introdurre o a cambiare le date delle scadenze nel vostro file delle date. Vengono chiesti un *ANNO*, un *MESE* un *GIORNO* e il *TESTO* del messaggio. L'*ANNO* è quello previsto in memoria, in caso contrario viene chiesto, come per l'opzione *STAMPA CALENDARIO, "CARICO DA DISCO?"*. Anche qui una freccia verso l'alto fa tornare al menu principale. Questa opzione non permette di introdurre date non valide come 32 Marzo o 31 Novembre.
Alla richiesta di *TESTO* se c'è già una scadenza in memoria per l'anno-mese-giorno impostato, il testo viene mostrato sotto la linea di puntini sulla quale si introdurrà quello nuovo. Dando un *RETURN* il vecchio testo sarà conservato, mentre introducendo un asterisco (★), lo si cancellerà.
Si può scrivere qualsiasi frase purchè non ecceda i 19 caratteri e non vi sia uno slash (/) al suo interno. Infatti la presenza di uno slash indica al programma che intendete avere due scadenze per quella data. In questo caso ogni messaggio non può eccedere i 9 caratteri, mentre se vi è qualche messaggio corrente (per default), esso verrà cancellato dall'introduzione del nuovo. Finita la fase di impostazione-modifica del testo, digitando una freccia verso l'alto (↑) verrà chiesto *"VUOI SALVARE IL FILE SU DISCO?"* e rispondendo "*S*" verrà creato un file sul disco denominato *CAL. FILE-19XX* in cui *XX* è l'anno su cui si stava lavorando. Questo file può naturalmente essere richiamato in seguito per correzioni o per la stampa.

Imposta giorni festivi

Questa opzione si usa per impostare tutte o alcune delle festività di un dato anno nel file delle scadenze. Se le festività introdotte si riferiscono a un anno precedente il 1987, le date relative vengono mostrate dal programma quando si digita la festività.
Per il 1987 o oltre verrà presentato il nome della festività, ma il giorno in cui questa cade dovrà essere introdotto da tastiera. In ogni caso giorni come Capodanno e Natale, che cadono sempre allo stesso giorno del mese, recheranno sempre la data di quel giorno.
Innanzitutto viene chiesto *l'ANNO*. Come per l'opzione "imposta-modifica date importanti" se l'anno non coincide con quello del file di date in memoria viene chiesto se si desidera caricarlo da disco. Viene mostrato il nome di una festività, poi il suo mese e se ne chiederà il giorno o, se si riferisce a un anno precedente il 1987, esso verrà mostrato direttamente. Quindi verrà chiesto *OK?*. Se si vuole che questa festività appaia sul calendario si risponde *S*. Se questo giorno cade su uno per il quale si è già salvato un messaggio, ne verrà mostrato il testo e verrà chiesto se lo si vuole sostituire e si risponde *S* oppure *N*. Se viene chiesto un giorno e non si desidera che quella festa appaia sul calendario, si digita un asterisco (★). Una volta impostate tutte le festività o se si digita una freccia in alto, o viene chiesto se si desidera salvare questo file di date, come nell'opzione imposta-modifica date importanti.

Lista date importanti

Questa opzione serve a visionare sul monitor o stampare un file di date con sca-

denze. Si risponde alla domanda relativa all'*ANNO, DAL* e *AL MESE* come prima. Se è necessario viene chiesto se si vuole caricare un file di date da disco.
Poi viene chiesto di specificare *"STAMPANTE O VIDEO?"* cui si risponde *S* per stampare, con la scheda nel connettore 1 dell'Apple, oppure *V*.

Fine

Questa opzione serve a lasciare il programma. Se non si è salvato su disco il file sul quale si è lavorato, conviene farlo ora (andando in imposta-modifica date importanti e digitando una freccia in alto per tornare). Poichè *FINE* non salva il file automaticamente vi viene chiesto se ne siete sicuri, quindi il programma vi saluta dicendo *"CIAO"*.

Come funziona il programma

Il programma del calendario parte inizializzando alcune importanti variabili:

N$ (12,31)	*Tiene il testo per il mese, giorno.*
ME$ (12)	*Tiene il nome dei mesi.*
L$	*CHR$ (124) = 1 (trattino verticale per le caselle dei giorni).*
D$	*CHR$ (4) = Control-D, comandi DOS.*
GI$	*Stringa di due cifre per il numero di giorni di ciascun mese.*
SLOT	*Il connettore in cui è inserita la scheda della stampante nell'Apple a partire da 1*

Vengono definite due altre funzioni: una è un controllo in modulo 4 per il controllo degli anni bisestili, l'altra è in modulo 7 per controllare i giorni della settimana. Viene quindi mostrata l'intestazione con una *GOSUB 4560*.
L'asterisco in movimento è creato con cicli *FOR* che stampano alternativamente uno spazio e un asterisco definendo una specie di cornice. Viene controllato durante l'operazione che non ci sia alcun segnale dalla tastiera, e nel caso ci sia si ha un *RETURN* dalla subroutine.
Alla linea *180* viene effettuata una *GOSUB 4990* che attiva una routine in linguaggio macchina dal manuale di riferimento Apple (pag. 136) per la gestione dei problemi con la catasta operativa causati dagli errori trattati dal comando *ONERR GOTO*. In questo programma vi sono molti cicli *FOR* e *GOSUB* ciascuno dei quali indirizza dei ritorni sulla catasta. Se un errore viene ripetuto più volte in una riga, la catasta si scompiglia. Può accadere per esempio che ripetendo alla domanda *"CARICO DA DISCO?"* un *"S"* per diverse volte in una riga, si ottenga un errore di *OUT OF MEMORY* poichè il programma cerca di caricare diverse volte un file non esistente. Per un paio di volte la catasta non ne risente e l'errore verrà gestito correttamente dalla routine di gestione degli errori *(FILE NOT FOUND* - non trovato), ma poi il sistema potrebbe fare strane cose. Se si verifica un errore si può battere *GOTO 250* per ripartire senza perdere l'inizializzazione delle variabili. Infine quando il programma parte viene fatto un tentativo di caricare il file delle date del 1981 dal disco. Il comando *ONERR* alla linea *200* protegge dal fatto che questo file sia indefinito.

La routine di trattamento degli errori che parte alla linea *4210* è appunto intesa per questo tipo di errori sull'input e l'output da disco.
Il *MENU* viene mostrato a partire dalla linea *250*. Alla linea *390* viene eseguito un *GOTO* anzichè una *GOSUB* perchè la catasta si stava riempiendo di *GOSUB* e cicli *FOR* nidificati per la stampa del calendario. Questa è anche la ragione per cui molti cicli sono fissati con valori anzichè con cicli *FOR/NEXT*. Per via del *GOTO* ogni opzione deve eseguire un *GOTO 250* quando ha finito il suo lavoro.
La routine di stampa calendario parte alla linea *400*. Affinchè si abbia un video formattato, tutte le richieste vengono poste insieme (linee da *420* a *450*) mentre le domande vengono poste poi una alla volta. Alla linea *640* se l'anno impostato non coincide con quello in memoria, viene eseguita un *GOSUB 3860* per chiedere *"CARICO DA DISCO"*. Alla linea viene abilitato l'*ONERR* nel caso che il file richiesto non sia sul disco. Si può notare che il file delle date è semplice.
12 (mesi) ★ 31 (giorni) = 372 campi sequenziali di dati, in maggioranza nulli, ma alcuni con del testo. Il nome del file è *"CAL. FILE-19XX"* dove *XX* è la seconda metà dell'anno scelto.
Ogni qualvolta si introduce il nome di un mese, una *GOSUB 1280* viene eseguita per controllare la corretta sintassi della sigla confrontandola con la stringa delle abbreviazioni alla linea *1310*. In risposta si ottiene il numero del mese. Se l'input non è valido *ERR* viene posto a 1.

I giorni della settimana

La successiva subroutine viene chiamata alla linea *1770* con un salto alla linea *1840*. Questa è una delle routine più importanti perchè calcola il giorno della settimana del primo giorno del mese dell'anno che si è scelto.
La variabile *KK* inizia da 5 perchè il primo Gennaio 1981 giorno di inizio del calendario, era un Giovedì, il quinto della settimana partendo dalla Domenica. Quindi viene calcolato il numero di anni tra il 1981 e l'anno impostato da tastiera, questo numero viene moltiplicato per 365 e aggiunto a *KK* (si prevedono gli anni bisestili con la *GOSUB 2060* alla linea *830*). Il calcolo degli anni bisestili rende la variabile *BISE = 0* oppure *1*, a seconda che sia occorso l'anno bisestile oppure no (vale a dire *MOD 4 = 0* mentre *MOD 100* non è *= 0*). Poi vengono calcolati i mesi tra Gennaio e quello impostato per sapere il numero di giorni da aggiungere (ricavati dalla stringa *GI$* a *KK* (linea *860* fino a *890*), anche qui considerando che Febbraio ha un giorno extra dovuto agli anni bisestili.
Infine *KK* (il giorno) viene calcolato in *MOD 7* per sapere che giorno della settimana sia (linea *900*).

Stampa del calendario

Fornite queste informazioni il calendario è pronto per essere stampato. Il programma ricicla entro *DAL* e *AL MESE* nelle linee da *930 a 1030* per stampare ciascun mese. Per stampare il mese precedente e il seguente in calce a ciascun calendario viene eseguita una *GOSUB 1740*. La stampa effettiva parte alla linea *1040*. La maggior parte delle istruzioni sono auto esplicative. Alla linea *1220* la variabile *GIO* (giorno) da *KK* viene negata e vi viene aggiunto 2. Così se fosse 3 (= Martedì), diventerebbe —1. Viene incrementata per ciascun giorno e quando diventa > 0 allora partono i numeri dei giorni, in modo che il primo giorno cada nel corretto giorno della settimana (Martedì in questo caso). La stampa è controllata da tre loop *FOR* nidificati; il primo alla linea *1230* per la stampa delle linee orizzontali di divisione, il secondo dalla linea *1240* per i sei trattini verticali per ogni casella, e il terzo per la stampa dei sette giorni orizzontalmente.
L'ultimo è costituito in reatà di tre loop, uno usato per *RR = 1* (prima linea) per stampare il numero del giorno (linea *1260*), uno usato per *RR = 2* fino a 4 per stampare gli spazi (linea *1350*) e uno usato per *RR=5* e 6 per stampare qualsiasi testo corrispondente alla data contenuto nell'ordinamento *N$* e individuato dal mese e dal giorno (linea *1390*). L'ultimo loop se trova un valore non nullo in *N$* esegue una *GOSUB 1460* per stampare le note del calendario. Questa routine divide il testo in due linee se è troppo lungo e ricerca uno slash incorporato che gli indichi due scadenze separate. Viene poi stampato il testo appropriato a seconda che *RR* sia sulla linea 5 oppure 6. Quando il calendario è finito, la parte in calce viene stampata con una *GOSUB 1740*. La routine alla linea *780* serve per individuare dove cade il primo giorno del mese prece-

Figura 1 - La stampa di un mese mostra il calendario completo e segnala le date importanti precedentemente preregistrate. Per una migliore pianificazione, in coda vengono anche stampati il mese precedente e quello successivo, in questo caso Marzo e Maggio.

```
ELENCO DELLE DATE: 1983
DA APRILE A APRILE

APRILE
  2     PULIZIE / PRIMAV.
  3     PASQUA
  4     PASQUETTA
  7     CORSO DI / INGLESE
  10     COMPLEAN / NO ZIA
  18     COMPRARE / BIT
  20     APPUNT. / DENTISTA
  24     GITA A / VENEZIA
  25     LIBERAZIONE
  27     STIPENDIO
```

Figura 2 - Un elenco delle date importanti inserite in un determinato mese, richiesto tramite il menu del programma Planel.

dente e del seguente, mentre i loop alle righe *1820*, *1870* e *1940* stampano tali mesi.

L'introduzione dei dati

IMPOSTA/MODIFICA DATE IMPORTANTI inizia alla linea *2100*. Qualsiasi data impostata viene controllata alle linee da *2280* a *2300* per accertare che sia valida. La routine *IMPOSTA GIORNI FESTIVI* inizia alla linea *2480* e si basa sull'anno che si sta considerando. Se questo è precedente al 1987 viene fatto un conto (linea *2580*) per generare un numero da usarsi per posizionare il puntatore nell'istruzione *DATA* nel punto giusto. Altrimenti questo numero è zero. Viene eseguito un loop per leggere le istruzioni *DATA* che partono alla linea *2950* e che sono strutturate in questo modo.

Ci sono 17 scadenze per anno, ciascuna è un giorno concatenato con una festività la quale è a sua volta concatenata a un mese. Se il giorno è sconosciuto (per anni successivi al 1986), il giorno non è concatenato.

Via via che vengono lette le righe di *DATA*, vengono mostrati i giorni e i mesi (se sono noti) e viene chiesto all'utente se vanno bene. Se la risposta è affermativa, la festività viene archiviata nell'ordinamento *N$* (mese, giorno).

Se il giorno è zero (linea *2610*), viene attesa una data esatta da parte dell'utente. Se il giorno è noto il programma procede sino alla linea *2770*.

Altrimenti continua alla linea *2620*. Quando l'utente ha finito l'impostazione delle festività gli viene chiesto *"VUOI SALVARE IL FILE SU DISCO?"*. Se la risposta è "S" viene eseguita una *GOSUB 6000* che esegue la memorizzazione. In questa routine è anche attivato il flag di *ONERR*.

La routine di *LISTA DATE IMPORTANTI* che parte alla linea *3160* è breve e semplice. Dopo aver ottenuto *l'ANNO* e il *DAL/AL MESE* (convertito in numeri), chiede all'utente se preferisce stampare o visionare sullo schermo. Poi usa l'ordinamento *N$* per stampare le date importanti.

Da ultimo *FINE* parte alla linea *4120*. Viene chiesto "*SEI SICURO?*" e se lo siete vi viene detto "*CIAO*". Questa routine non salva automaticamente nessun file.

Modifiche al programma

Si può modificare questo programma in diversi modi. Per adattarsi alla configurazione di cui si dispone si può modificare il numero del connettore cui è collegata la stampante (linea *30*). Lo stesso calendario può venire modificato per eliminare, per esempio, gli asterischi che compaiono sopra e sotto il mese nella stampa, o per stampare il vostro nome su ogni foglio o per cambiare le sbarrette verticali usate per le caselle. In questo caso fate attenzione! Bisogna che il numero di caratteri sia lo stesso per mantenere la griglia del calendario.

Si può modificare anche la manipolazione dei testi se per esempio si desiderano tre linee di testo anziché due. Si cambia in questo caso la linea *1340* e da *1470* a *1620* e il controllo sulla lunghezza del testo introdotto: ma attenzione che non è un compito banale.

La routine di *IMPOSTA FESTE IMPORTANTI* può essere modificata facilmente aggiungendone altre di propria scelta (devono avere però il formato giorno - nome della festa - anno, come le altre già inserite in *DATA*). Bisogna inoltre cambiare alle linee *2580* il numero che moltiplica l'argomento dell'istruzione *THEN*, e alla linea *2600* il numero che indica le iterazioni della variabile nel loop *FOR/NEXT* mettendoci il nuovo numero di ricorrenze.

Per coloro che hanno la stampante Silentype bisogna cambiare alcune linee come indicato sotto:

```
1640 PRINT SPC (S); LEFT $
("************". L+3)
1660 PRINT SPC (S-11); "+";
1690 PRINT SPC (S); LEFT $
("************", L +3)
```

Conclusione

Il miglior modo di accrescere la propria abilità nella programmazione è di leggere le istruzioni dettate da altri e modificarle. Ciò permette inoltre di personalizzare un programma e di renderlo più facile da usare e più divertente. Quindi buon divertimento e stampante tanti calendari con *Planel*.

```
10  REM      "*******************"
20  REM      "*                 *"
30  REM      "*      PLANEL     *"
40  REM      "*                 *"
50  REM      "*      BIT 83     *"
60  REM      "*                 *"
70  REM      "*******************"
80  REM
90  REM
100  DIM N$(12,31),ME$(12)
110 L$ =  CHR$ (124):D$ =  CHR$ (4)
120 GI$ = "312831303130313130313031":SL
OT = 1
130  FOR I = 1 TO 12: READ Z$:ME$(I) =
Z$: NEXT I
140  DATA   GENNAIO,FEBBRAIO,MARZO,APRI
LE,MAGGIO,GIUGNO,LUGLIO,AGOSTO,SETTEMBR
E,OTTOBRE,NOVEMBRE,DICEMBRE
150  DEF  FN M(X) = ((X / 4 -  INT (X /
 4)) * 4): DEF  FN SE(X) = ((X / 7 - I
NT (X / 7)) * 7)
160  GOSUB 4560: REM   TITOLO PAGINA
170  HOME : VTAB 10: HTAB 11: PRINT "<A
TTENDERE...>"
180  GOSUB 4990
190 AC = 1981:NOO = 1
200  ONERR  GOTO 4210
210 SAL = 0
220  GOSUB 3870
230  POKE 216,0
240 NOO = 0
250  REM   ** MENU **
260  HOME : PRINT : PRINT
270  INVERSE
280  PRINT "  PLANEL  "; : HTAB 24: PRIN
T "BIT 1983"
290  NORMAL
300  VTAB 8
310  PRINT "1) STAMPA CALENDARIO"
320  VTAB 10: PRINT "2) IMPOSTA/MODIFIC
A DATE IMPORTANTI"
330  VTAB 12: PRINT "3) IMPOSTA GIORNI
FESTIVI"
340  VTAB 14: PRINT "4) LISTA DATE IMPO
RTANTI"
350  VTAB 16: PRINT "5) FINE"
360  VTAB 22: INVERSE : PRINT "QUALE ?"
; : NORMAL
370  INPUT " ";Z$:I =  VAL (Z$)
380  IF I < 1 OR I > 5 THEN  PRINT  CHR
$ (7): GOTO 260
390  ON I GOTO 400,2100,2480,3160,4120
400  REM   ** STAMPA CALENDARIO **
410 Z$ = "STAMPA CALENDARIO": GOSUB 407
0
420  VTAB 8: HTAB 1: PRINT "DAL MESE (G
EN): "; : CALL  - 868
430  VTAB 10: HTAB 1: PRINT "AL  MESE (
DIC): "; : CALL  - 868
440  VTAB 12: HTAB 1: PRINT "ANNO (";AC
;"): "; : CALL  - 868
450  VTAB 14: HTAB 1: PRINT "     OK? (
S): "; : CALL  - 868
460  VTAB 8: HTAB 19: CALL  - 868: INPU
T "";Z$
470  IF Z$ = "^" THEN 250
480  IF Z$ = "" THEN Z$ = "GEN": VTAB 8
: HTAB 19: PRINT "GEN"
490  GOSUB 700: IF ERR THEN 460
500 F =  VAL (Z$)
510  VTAB 10: HTAB 19: CALL  - 868: INP
UT "";Z$
520  IF Z$ = "" THEN Z$ = "DIC": VTAB 1
0: HTAB 19: PRINT "DIC"
530  GOSUB 700: IF ERR THEN 510
540 T =  VAL (Z$)
550  IF F > T THEN  PRINT  CHR$ (7); : G
OTO 460
560  VTAB 12: HTAB 14: CALL  - 868: INP
UT "";A$
570  IF A$ = "^" THEN 250
580  IF A$ = "" THEN A$ =  STR$ (AC): V
TAB 12: HTAB 14: PRINT A$
590 P =  VAL (A$): IF P < 1981 OR P > 2
000 THEN  PRINT  CHR$ (7); : GOTO 560
600  VTAB 14: HTAB 15: CALL  - 868: INP
UT "";Z$
610  IF Z$ < > "" AND  LEFT$ (Z$,1) <
 > "S" THEN 250
620  VTAB 14: HTAB 15: PRINT "SI"
630  REM   ** CARICA DA DISCO ? **
640  IF AC < > P THEN AC = P: GOSUB 38
00
650  GOSUB 780: REM   ** CERCA IL PRIMO
 GIORNO DEL MESE **
660  GOTO 930
670  PRINT D$;"PR#0"
680  GOTO 250
690  REM
700  REM   ** MESE ERRATO **
710 ERR = 0
720  FOR I = 1 TO 12
730  IF Z$ =  MID$ ("GENFEBMARAPRMAGGIU
LUGAGOSETOTTNOVDIC",I * 3 - 2,3) THEN 7
60
740  NEXT I
750  PRINT  CHR$ (7); :ERR = 1: RETURN
760 Z$ =  STR$ (I): RETURN
770  REM
780  REM   ** LOOP NEI MESI **
790 KK = 5
800 AN = P - 1981 - 1: IF AN < 0 THEN 8
50
810  FOR I = 0 TO AN
820 KK = KK + 365
830 AA = 1981 + I: GOSUB 2060:KK = KK +
 BISE
840  NEXT
850 MS = F - 1: IF MS < 1 THEN 900
860  FOR I = 1 TO MS
870 KK = KK +  VAL ( MID$ (GI$,I * 2 -
1,2))
880  IF I = 2 THEN AA = P: GOSUB 2060:K
K = KK + BISE: REM    ** FEB IN ANNI B
ISE**
890  NEXT
900 GIO =  INT ( FN SE(KK) + .5): REM
  ** GIORNO DELLA SETTIMANA DEL PRIMO G
IORNO DEL MESE
910  IF GIO = 0 THEN GIO = 7
920  RETURN
930 M = F - 1
940 M = M + 1: IF M > T THEN 1030
950  GOSUB 1040
960 KH = KK: GOSUB 1740: REM   ** ULTIM
O E PROSSIMO MESE
970 KK = KH +  VAL ( MID$ (GI$,M * 2 -
1,2))
980  IF M = 2 THEN AA = P: GOSUB 2060:K
K = KK + BISE
990 GIO =  INT ( FN SE(KK) + .5)
1000  IF GIO = 0 THEN GIO = 7
1010  PRINT : PRINT : PRINT : PRINT : P
RINT : PRINT : REM   ** PROSSIMA PAGINA
 **
1020  GOTO 940
1030  GOTO 670
1040  REM   ** STAMPA **
1050  PRINT : PRINT D$;"PR#";SLOT
1060  PRINT "80N"
1070  PRINT : PRINT : PRINT
1080 M$ = ME$(M):L =  LEN (M$) * 2:ND =
  VAL ( MID$ (GI$,M * 2 - 1,2))
1090  IF M = 2 THEN AA = P: GOSUB 2060:
ND = ND + BISE
```

Seguito listato Planel.

```
1100 S = (78 - L - 3) / 2
1110  PRINT  TAB( S); LEFT$ ("*********
************",L + 3)
1120  PRINT "   APPLE ]"; CHR$ (91);"+"
;
1130  PRINT  TAB( S - 12),"* ";
1140  FOR I = 1 TO L / 2: PRINT  MID$ (
M$,I,1);" ";: NEXT I: PRINT "*";
1150  PRINT  SPC( 67 - S - L);P
1160  PRINT  TAB( S); LEFT$ ("*********
************",L + 3)
1170  PRINT : FOR I = 1 TO 78: PRINT "=
";: NEXT
1180  PRINT : PRINT L$;"     DOM    ";L$;
"    LUN    ";
1190  PRINT L$;"    MAR    ";L$;"    MER
    ";L$;"    GIO    ";L$;
1200  PRINT "    VEN    ";L$;"    SAB
";L$
1210  FOR I = 1 TO 78: PRINT "=";: NEXT
 : PRINT
1220 GIO =  - GIO + 2
1230  FOR R = 1 TO 6
1240  FOR RR = 1 TO 6
1250  PRINT L$;: IF RR > 1 THEN 1340
1260  FOR I = 1 TO 7
1270  IF GIO < 1 OR GIO > ND THEN  PRIN
T "  ";: GOTO 1300
1280  IF GIO < 10 THEN  PRINT " ";
1290  PRINT GIO;
1300 GIO = GIO + 1
1310  PRINT "        ";L$;
1320  NEXT I
1330  GOTO 1630
1340  IF RR > 4 THEN 1390
1350  FOR I = 1 TO 7
1360  PRINT "          ";L$;
1370  NEXT I
1380  GOTO 1630
1390  FOR D = GIO - 7 TO GIO - 1
1400  IF D < 1 OR D > ND THEN SP = 10:
GOTO 1430
1410 SP = 10
1420  IF N$(M,D) <  > "" THEN  GOSUB 14
60:SP = LFT
1430  PRINT  SPC( SP);L$;
1440  NEXT D
1450  GOTO 1630
1460  REM   ** NOTE SUL CALENDARIO **
1470 Z$ = N$(M,D)
1480 L = 0
1490 L = L + 1: IF L >  LEN (Z$) THEN 1
520
1500  IF  MID$ (Z$,L,1) = "/" THEN 1610

1510  GOTO 1490
1520  IF  LEN (Z$) < 10 THEN LFT = 10:
IF RR = 6 THEN  PRINT Z$;:LFT = 10 - L
EN (Z$)
1530  IF  LEN (Z$) < 10 THEN  RETURN
1540 L =  LEN (Z$) / 2:LFT = 10 - L
1550  IF L <  >  INT (L) THEN 1580
1560  IF RR = 5 THEN  PRINT  LEFT$ (Z$,
L);"-";:LFT = LFT - 1: RETURN
1570  PRINT  RIGHT$ (Z$,L);: RETURN
1580  IF RR = 5 THEN  PRINT  LEFT$ (Z$,
L);"-";: RETURN
1590  PRINT  RIGHT$ (Z$,L + 1);: RETURN

1600  REM   ** 2 LINEE DI DATI (/) **
1610  IF RR = 5 THEN  PRINT ". "; LEFT$
(Z$,L - 1);:LFT = 10 - L: RETURN
1620  PRINT "."; RIGHT$ (Z$, LEN (Z$) -
 L);:LFT = 9 - LEN (Z$) + L: RETURN
1630  PRINT
1640  NEXT RR
1650  PRINT L$;
1660  FOR I = 1 TO 7
1670  PRINT "----------";
1680  IF I < 7 THEN  PRINT "+";
1690  IF I = 7 THEN  PRINT L$
1700  NEXT I
1710  NEXT R
1720  RETURN
1730  REM
1740  REM   ** STAMPA L'ULTIMO E IL PRO
SSIMO MESE **
1750  IF M = 1 AND P = 1981 THEN G1 =
- 2 + 2:N1 = 31:: GOTO 1790: REM    DIC
 1980
1760 F = M - 1: IF F < 1 THEN F = 12:P
= P - 1
1770  GOSUB 780: IF F = 12 THEN P = P +
 1
1780 G1 =  - GIO + 2:N1 =  VAL ( MID$ (
GI$,F * 2 - 1,2)): IF F = 2 THEN AA = P
: GOSUB 2060:N1 = N1 + BISE
1790 F = M + 1: IF F > 12 THEN F = 1:P
= P + 1
1800  GOSUB 780: IF F = 1 THEN P = P -
1
1810 G2 =  - GIO + 2:N2 =  VAL ( MID$ (
GI$,F * 2 - 1,2)): IF F = 2 THEN AA = P
: GOSUB 2060:N2 = N2 + BISE
1820  FOR I = 1 TO 6
1830  IF I = 3 THEN  PRINT "******** PL
ANEL *******";: GOTO 1860
1840  IF I = 4 THEN  PRINT "********* B
IT *********";: GOTO 1860
1850  PRINT  SPC( 23);
1860  PRINT  SPC( 4);
1870  FOR J = 1 TO 7
1880  IF G1 < 1 THEN  PRINT "   ";:G1 =
 G1 + 1: GOTO 1920
1890  IF G1 > N1 THEN  PRINT "   ";: GO
TO 1920
1900  IF G1 < 10 THEN  PRINT " ";
1910  PRINT " ";G1;:G1 = G1 + 1
1920  NEXT J
1930  PRINT  SPC( 5);
1940  FOR J = 1 TO 7
1950  IF G2 < 1 THEN  PRINT "   ";:G2 =
 G2 + 1: GOTO 1990
1960  IF G2 > N2 THEN  PRINT "   ";: GO
TO 1990
1970  IF G2 < 10 THEN  PRINT " ";
1980  PRINT " ";G2;:G2 = G2 + 1
1990  NEXT J
2000  PRINT
2010  NEXT I
2020  PRINT  SPC( 35);ME$(M - 1 + 12 *
(M =1));
2030 I =  LEN (ME$(M - 1 + 12 * (M = 1)
))
2040  PRINT  SPC( 60 - 35 - I);ME$(M +
1 - 12 * (M = 12))
2050  RETURN
2060  REM   ** ANNO BISE  **
2070 BISE = 0
2080  IF  INT ( FN M(AA) + .5) = 0 AND
AA / 100 <  >  INT (AA / 100) THEN BISE
 = 1
2090  RETURN
2100  REM   ** IMPOSTA/MODIFICA DATE IM
PORTANTI **
2110 Z$ = "IMPOSTA/MODIFICA DATE IMPORT
ANTI": GOSUB 4070:L = 0
2120  VTAB 8: HTAB 1: CALL  - 868: PRIN
T "ANNO (";AC;"):"
2130  VTAB 12: HTAB 1: CALL  - 868: PRI
NT "MESE:"
2140  VTAB 14: HTAB 1: CALL  - 868: PRI
NT "GIORNO:"
2150  VTAB 16: HTAB 1: CALL  - 868: PRI
```

```
NT "TESTO:"
2160  IF L THEN 2220
2170  VTAB 8: HTAB 14: CALL  - 868: INP
UT "";A$
2180  IF A$ = "" THEN A$ =  STR$ (AC):
VTAB 8: HTAB 14: PRINT A$;
2190  IF A$ = "^" THEN 250
2200 P =  VAL (A$): IF P < 1981 OR P >
2000 THEN  PRINT  CHR$ (7);: GOTO 2170
2210  IF P <  > AC THEN  GOSUB 3800: AC
= P
2220  VTAB 12: HTAB 6: CALL  - 868: INP
UT "";Z$
2230  IF Z$ = "^" THEN 2450
2240  GOSUB 700: IF ERR THEN 2220
2250 M =  VAL (Z$)
2260  VTAB 14: HTAB 8: CALL  - 868: INP
UT "";Z$
2270  IF Z$ = "^" THEN 250
2280 GIO =  VAL (Z$): IF GIO = 0 THEN
PRINT  CHR$ (7);: GOTO 2260
2290 D =  VAL ( MID$ (GI$,M * 2 - 1,2))
: IF GIO > D AND M <  > 2 THEN  PRINT
CHR$ (7);: GOTO 2260
2300  IF M = 2 THEN AA = P: GOSUB 2060:
 IF GIO > D + BISE THEN  PRINT  CHR$ (7
);: GOTO 2260
2310 A$ = "": IF N$(M,GIO) <  > "" THEN
 A$ = N$(M,GIO): VTAB 17: HTAB 7: CALL
 - 868: PRINT "(";A$;")";
2320  VTAB 16: HTAB 7: CALL  - 868: PRI
NT "..................";: HTAB 7: INPUT
 "";Z$
2330  IF Z$ = "" AND A$ <  > "" THEN  C
ALL  - 868: VTAB 16: HTAB 7: PRINT A$:
GOTO 2420
2340  IF Z$ = "*" THEN  CALL  - 868: PR
INT "<CANCELLATO>";:N$(M,GIO) = "": GOT
O 2420
2350  IF Z$ = "^" THEN 2450
2360  IF Z$ = "" THEN 2420
2370  IF  LEN (Z$) > 18 THEN  PRINT  CH
R$ (7);: GOTO 2310
2380  FOR I = 1 TO  LEN (Z$): IF  MID$
(Z$,I,1) = "/" THEN 2400
2390  NEXT : GOTO 2410
2400  IF I > 10 OR  LEN (Z$) - I > 9 TH
EN  PRINT  CHR$ (7);: GOTO 2310
2410 N$(M,GIO) = Z$
2420  FOR I = 1 TO 750: NEXT I
2430 L = 1
2440  VTAB 11: CALL  - 958: GOTO 2130
2450  VTAB 20: CALL  - 958: INPUT "VUOI
 SALVARE SU DISCO ? ";Z$
2460  IF  LEFT$ (Z$,1) = "S" THEN  GOSU
B 3600
2470  GOTO 250
2480  REM   ** IMPOSTA GIORNI FESTIVI
**
2490 Z$ = "IMPOSTA GIORNI FESTIVI": GOS
UB 4070
2500  VTAB 8: HTAB 1: PRINT "ANNO (";AC
;") :"
2510  VTAB 12: HTAB 1: PRINT "GIORNI FE
STIVI :"
2520  VTAB 14: HTAB 5: PRINT "MESE :";:
 HTAB 28: PRINT "GIORNO :"
2530  VTAB 8: HTAB 14: CALL  - 868: INP
UT "";A$
2540  IF A$ = "" THEN A$ =  STR$ (AC):
VTAB 8: HTAB 14: PRINT A$;
2550  IF A$ = "^" THEN 250
2560 P =  VAL (A$): IF P < 1981 OR P >
2000 THEN  PRINT  CHR$ (7);: GOTO 2530
2570  IF P <  > AC THEN  GOSUB 3800: AC
= P
2580 R = 0: IF P < 1987 THEN R = (6 + P
2590  RESTORE : FOR I = 1 TO 12 + R: RE
AD Z$: NEXT
2600  FOR I = 1 TO 13
2610  READ Z$: IF  VAL (Z$) > 0 THEN 27
70
2620 M =  VAL ( RIGHT$ (Z$,2)):Z$ =  LE
FT$ (Z$, LEN (Z$) - 2)
2630  VTAB 12: HTAB 10: PRINT Z$;
2640  VTAB 14: HTAB 13: PRINT ME$(M);
2650  VTAB 14: HTAB 34: CALL  - 868: IN
PUT "";A$
2660  IF A$ = "^" THEN 2920: REM   ** V
UOI SALVARE ? **
2670  IF A$ = "*" THEN 2890: REM   ** P
ROSSIMO **
2680 GIO =  VAL (A$): IF GIO = 0 THEN
PRINT  CHR$ (7);: GOTO 2650
2690 D =  VAL ( MID$ (GI$,M * 2 - 1,2))
: IF GIO > D AND M <  > 2 THEN  PRINT
CHR$ (7);: GOTO 2650
2700  IF M = 2 THEN AA = P: GOSUB 2060:
 IF GIO > D + BISE THEN  PRINT  CHR$ (7
);: GOTO 2650
2710  IF N$(M,GIO) = "" THEN 2880: REM
  ** FILE **
2720  VTAB 20: HTAB 1: CALL  - 868: PRI
NT "SOSTITUISCI ";N$(M,GIO)
2730  PRINT "   CON ";Z$;: INPUT "
OK ? ";B$
2740  IF  LEFT$ (B$,1) = "S" THEN 2880:
 REM   ** FILE **
2750  VTAB 21: CALL  - 868: PRINT "
<NON SOSTITUITO>";
2760  GOTO 2890: REM   ** TESTO **
2770 M =  VAL ( RIGHT$ (Z$,2)):GIO =  V
AL ( LEFT$ (Z$,2)):Z$ =  MID$ (Z$,3, LE
N (Z$) - 4)
2780  VTAB 12: HTAB 19: PRINT Z$;
2790  VTAB 14: HTAB 13: PRINT ME$(M);
2800  HTAB 36: PRINT GIO
2810  IF N$(M,GIO) = "" OR Z$ = N$(M,GI
O) THEN 2840
2820  VTAB 20: HTAB 1: CALL  - 868: PRI
NT "VUOI SOSTITUIRE ";N$(M,GIO)
2830  PRINT "CON ";: INVERSE : PRINT Z$
;: NORMAL
2840  INPUT " OK ?(S/N) ";B$
2850  IF B$ = "^" THEN 2920
2860  IF  LEFT$ (B$,1) = "S" OR B$ = ""
 THEN 2880: REM   ** FILE **
2870  VTAB 21: CALL  - 868: PRINT "
<NESSUN FILE>";: FOR J = 1 TO 500: NEXT
 J: GOTO 2890: REM   ** PROSSIMO **
2880 N$(M,GIO) = Z$
2890  VTAB 10: CALL  - 958: VTAB 12: HT
AB 1: PRINT "GIORNI FESTIVI:"
2900  VTAB 14: HTAB 5: PRINT "MESE:";:
HTAB 28: PRINT "GIORNO:"
2910  NEXT I
2920  VTAB 20: CALL  - 958: INPUT " VUO
I SALVARE IL FILE SU DISCO ? ";Z$
2930  IF  LEFT$ (Z$,1) = "S" THEN  GOSU
B 3600
2940  GOTO 250
2950  REM   ** >1986 **
2960  DATA   01CAPODANNO01, 14SAN VALEN
TINO02, PASQUA04, PASQUETTA04, 25LIBERAZIO
NE04, 01FESTA DEL LAVORO05, 12FESTA DELLA
 MAMMA05, 21 FESTA DEL PAPA'06
2970  DATA   15FERRAGOSTO08, 01OGNISSANT
I11, 08IMMACOLATA CONCEZIONE12, 25NATALE1
2, 26SANTO STEFANO12
2980  REM   ** 1981 **
2990  DATA   01CAPODANNO01, 14SAN VALEN
TINO02, 12PASQUA04, 13PASQUETTA04, 25LIBER
AZIONE04, 01FESTA DEL LAVORO05, 12FESTA D
ELLA MAMMA05, 21FESTA DEL PAPA'06
3000  DATA   15FERRAGOSTO08, 01OGNISSANT
```

Planel, un supporto agli smemorati

Seguito listato Planel.

```
I11,08IMMACOLATA CONCEZIONE12,25NATALE1
2,26SANTO STEFANO12
3010  REM  ** 1982 **
3020  DATA   01CAPODANNO01,14SAN VALENT
INO02,11PASQUA04,12PASQUETTA04,25LIBERA
ZIONE04,01FESTA DEL LAVORO05,12FESTA DE
LLA MAMMA05
3030  DATA    21FESTA DEL PAPA'06,15FERR
AGOSTO08,01OGNISSANTI11,08IMMACOLATA CO
NCEZIONE12,25NATALE12,26SANTO STEFANO12
3040  REM  ** 1983 **
3050  DATA   01CAPODANNO01,14SAN VALENT
INO02,03PASQUA04,04PASQUETTA04,25LIBERA
ZIONE04,01FESTA DEL LAVORO05,12FESTA DE
LLA MAMMA05,21FESTA DEL PAPA'06,15FERRA
GOSTO08
3060  DATA          01OGNISSANTI11,08IMMA
COLATA CONCEZIONE12,25NATALE12,26SANTO
STEFANO12
3070  REM  ** 1984 **
3080  DATA        01CAPODANNO01,14SAN VAL
ENTINO02,22PASQUA04,23PASQUETTA04,25LIB
ERAZIONE04,01FESTA DEL LAVORO05,12FESTA
 DELLA MAMMA05
3090  DATA       21FESTA DEL PAPA'06,15FE
RRAGOSTO08,01OGNISSANTI11,08IMMACOLATA
CONCEZIONE12,25NATALE12,26SANTO STEFANO
12
3100  REM   ** 1985 **
3110  DATA          01CAPODANNO01,14SAN V
ALENTINO02,07PASQUA04,08PASQUETTA04,25L
IBERAZIONE04,01FESTA DEL LAVORO05,12FES
TA DELLA MAMMA05,21FESTA DEL PAPA'06
3120  DATA         15FERRAGOSTO08,01OGN
ISSANTI11,08IMMACOLATA CONCEZIONE12,25N
ATALE12,26SANTO STEFANO12
3130  REM  ** 1986 **
3140  DATA  01CAPODANNO01,14SAN VALEN
TINO02,30PASQUA03,31PASQUETTA03,25LIBER
AZIONE04,01FESTA DEL LAVORO05,12FESTA D
ELLA MAMMA05
3150  DATA    21FESTA DEL PAPA'06,15FER
RAGOSTO08,01OGNISSANTI11,08IMMACOLATA C
ONCEZIONE12,25NATALE12,26SANTO STEFANO1
2
3160  REM   ** ELENCO DELLE DATE **
3170 Z$ = "ELENCO DELLE DATE": GOSUB 40
70
3180  VTAB 8: HTAB 4: PRINT "ANNO (";AC
;"):"
3190  VTAB 10: HTAB 1: PRINT "DAL MESE
(GEN):"
3200  VTAB 12: HTAB 1: PRINT "AL MESE
(DIC):"
3210  VTAB 14: HTAB 1: PRINT "STAMPA O
VIDEO:"
3220  VTAB 8: HTAB 17: CALL  - 868: INP
UT "";A$
3230  IF A$ = "^" THEN 250
3240  IF A$ = "" THEN A$ =  STR$ (AC):
VTAB 8: HTAB 17: PRINT A$
3250 P =  VAL (A$): IF P < 1981 OR P >
2000 THEN  PRINT  CHR$ (7);: GOTO 3220
3260  IF P < > AC THEN  GOSUB 3800: AC
= P: REM   ** CARICA **
3270  VTAB 10: HTAB 17: CALL  - 868: IN
PUT "";Z$
3280  IF Z$ = "^" THEN 3160
3290  IF Z$ = "" THEN Z$ = "GEN": VTAB
10: HTAB 17: PRINT Z$
3300  GOSUB 700: IF ERR THEN 3270
3310 F =  VAL (Z$)
3320  VTAB 12: HTAB 17: CALL  - 868: IN
PUT "";Z$
3330  IF Z$ = "^" THEN 250
3340  IF Z$ = "" THEN Z$ = "DIC": VTAB
12: HTAB 17: PRINT Z$
3350  GOSUB 700: IF ERR THEN 3320
3360 T =  VAL (Z$)
3370  VTAB 14: HTAB 17: CALL  - 868: IN
PUT "";Z$
3380  IF Z$ = "^" THEN 250
3390  IF  LEFT$ (Z$,1) = "V" THEN 3430
3400  IF  LEFT$ (Z$,1) < > "S" THEN  P
RINT  CHR$ (7);: GOTO 3370
3410  PRINT D$;"PR#";SLOT
3420  PRINT : PRINT : PRINT
3430  HOME
3440 Z$ = "ELENCO DELLE DATE: " +  STR$
 (P)
3450  PRINT  TAB( 9);Z$
3460 Z$ = " DA " + ME$(F) + " A " + ME$
(T)
3470  PRINT  TAB( (40 -  LEN (Z$)) / 2)
;Z$
3480  PRINT : PRINT
3490  FOR M = F TO T
3500  PRINT : PRINT ME$(M)
3510  FOR D = 1 TO 31
3520  IF N$(M,D) = "" THEN 3540
3530  PRINT  TAB( 3);D; TAB( 7);N$(M,D)

3540  NEXT D
3550  NEXT M
3560  PRINT  CHR$ (12)
3570  PRINT D$;"PR#0"
3580  PRINT : PRINT "  <RETURN PER CONT
INUARE...";Z$
3590  GOTO 250
3600  REM   ** SALVA SU DISCO **
3610  ONERR  GOTO 4210
3620 SAL = 1
3630  PRINT " (";AC;: INPUT ") SEI SICU
RO ? ";B$
3640  IF  LEFT$ (B$,1) = "N" THEN 3770
3650  IF  LEFT$ (B$,1) < > "S" THEN  P
RINT  CHR$ (7);: GOTO 3630
3660 B$ = "CAL.FILE-" +  STR$ (AC)
3670  PRINT D$;"UNLOCK";B$
3680  PRINT D$;"OPEN";B$
3690  PRINT D$;"WRITE";B$
3700  FOR M = 1 TO 12
3710  FOR D = 1 TO 31
3720  PRINT N$(M,D)
3730  NEXT D
3740  NEXT M
3750  PRINT D$;"CLOSE";B$
3760  PRINT D$;"LOCK";B$
3770  POKE 216,0: REM   ** ERRORE **
3780  PRINT  CHR$ (7)
3790  RETURN
3800  REM   ** CARICA DA DISCO **
3810 SAL = 0
3820  VTAB 20: PRINT P;" IL FILE NON E'
IN MEMORIA"
3830  INPUT "CARICO DA DISCO ? ";B$
3840  IF  LEFT$ (B$,1) = "N" THEN 3980
3850  IF  LEFT$ (B$,1) < > "S" THEN  P
RINT  CHR$ (7);: GOTO 3830
3860  ONERR  GOTO 4210
3870 B$ = "CAL.FILE-" +  STR$ (P)
3880  PRINT D$;"OPEN";B$
3890  PRINT D$;"READ";B$
3900  FOR M = 1 TO 12
3910  FOR D = 1 TO 31
3920  INPUT N$(M,D)
3930  NEXT D
3940  NEXT M
3950  PRINT D$;"CLOSE";B$
3960  PRINT  CHR$ (7);: POKE 216,0: REM
 ** ERRORE **
3970  RETURN
3980  PRINT "  <CANCELLO TUTTI I VECCHI
```

```
FILE>"
3990  FOR M = 1 TO 12
4000  FOR D = 1 TO 31
4010 N$(M,D) = ""
4020  NEXT D
4030  NEXT M
4040  PRINT  CHR$ (7);
4050  FOR K = 1 TO 1000: NEXT K
4060  RETURN
4070  HOME : PRINT
4080 Z$ = " " + Z$ + " ": HTAB (40 -  L
EN (Z$)) / 2
4090  INVERSE : PRINT Z$: NORMAL
4100  PRINT
4110  RETURN
4120  REM   ** FINE **
4130 Z$ = "FINE": GOSUB 4070
4140  VTAB 5: CALL  - 868: INPUT "SEI S
ICURO ? ";Z$
4150  IF  LEFT$ (Z$,1) = "N" THEN 250
4160  IF  LEFT$ (Z$,1) <  > "S" THEN  P
RINT  CHR$ (7);: GOTO 170
4170  HOME : VTAB 10: HTAB 15: PRINT "*
 PLANEL *"
4180  VTAB 12: HTAB 15: PRINT "*** BIT
**"
4190  VTAB 15: HTAB 16: FLASH : PRINT "
CIAO !!": NORMAL : END
4200  PRINT : END
4210  REM   ** ERRORE **
4220 ERR =  PEEK (222)
4230 L =  PEEK (218) +  PEEK (219) * 25
6: CALL 768: REM     ** FISSA LA CATASTA
 **
4240  ON (ERR) GOTO 4250,4250,4250,4290
,4330,4410,4250,4460,4510
4250  REM      ** ERRORE DI  NON IDENTIF
ICAZIONE **
4260  PRINT : PRINT "ERRORE: "; CHR$ (7
);ERR
4270  PRINT "IN LINEA ";L; CHR$ (7)
4280  PRINT : END
4290  REM    ** PROTETTO IN SCRITTURA **

4300  PRINT "<PROTETTO IN SCRITTURA - I
NSERIRE NUOVO DISCO...>"
4310  INPUT " <E PREMERE RETURN: ";Z$
4320  GOSUB 3680: GOTO 250
4330  REM    ** FINE DATI **
4340  REM    ** ERRORE DI CARICAMENTO DA
 DISCO **
4350  IF SAL = 1 THEN 4250
4360  PRINT D$;"DELETE";B$
4370  VTAB 20: CALL  - 768
4380  IF NOO THEN 230
4390  PRINT "<NON TROVATO!>"
4400  POKE 216,0: GOSUB 3980: GOTO 250
4410  REM    ** FILE NON TROVATO **
4420  REM    ** ERRORE DI SALVATAGGIO SU
 DISCO **
4430  IF SAL = 0 THEN 4250
4440  GOSUB 3680: REM     ** SALTA UNLOCK
 **
4450  GOTO 250
4460  REM    ** I/O ERROR **
4470  PRINT "<I/O ERROR - INSERIRE UN N
UOVO DISCO>"
4480  INPUT " <RETURN PER CONTINUARE...
";Z$
4490  IF SAL = 1 THEN  GOSUB 3670: GOTO
 250
4500  IF SAL = 0 THEN  GOSUB 3880: GOTO
 250
4510  REM ** DISCO PIENO **
4520  PRINT D$;"DELETE";B$
4530  PRINT "<DISCO PIENO - INSERIRE UN
 NUOVO DISCO..."
4540  INPUT " <RETURN PER CONTINUARE...
";Z$
4550  GOSUB 3680: GOTO 250
4560  REM    ** INTESTAZIONE **
4570  TEXT : HOME :L = 25
4580  PRINT  CHR$ (7)
4590  VTAB 8: HTAB 9: PRINT "**********
***********"
4600  VTAB 12: HTAB 9: PRINT "*********
************"
4610  VTAB 9: FOR I = 1 TO 3
4620  HTAB 9: PRINT "*"; : HTAB 29: PRIN
T "*"
4630  NEXT I
4640  FLASH
4650  VTAB 10: HTAB 14: PRINT "  PLANEL
 "
4660  NORMAL
4670  VTAB 15: HTAB 13: PRINT "**    BIT
 **"
4680  POKE  - 16368,0
4690  VTAB 23: PRINT " <PREMERE UN TAST
O PER CONTINUARE...>"
4700  FOR I = 1 TO 2 STEP 0
4710  VTAB 8: FOR J = 9 TO 28
4720  HTAB J: PRINT " ";
4730  FOR D = 1 TO L: NEXT D
4740  HTAB J: PRINT "*";
4750  NEXT J
4760  IF  PEEK ( - 16384) > 127 THEN 49
60
4770  FOR J = 8 TO 11
4780  VTAB J: HTAB 29: PRINT " ";
4790  FOR D = 1 TO L: NEXT D
4800  HTAB 29: PRINT "*";
4810  NEXT J
4820  IF  PEEK ( - 16384) > 127 THEN 49
60
4830  VTAB 12: FOR J = 29 TO 10 STEP  -
 1
4840  HTAB J: PRINT " ";
4850  FOR D = 1 TO L: NEXT D
4860  HTAB J: PRINT "*";
4870  NEXT J
4880  IF  PEEK ( - 16384) > 127 THEN 49
60
4890  FOR J = 12 TO 9 STEP  - 1
4900  HTAB 9: VTAB J: PRINT " ";
4910  FOR D = 1 TO L: NEXT D
4920  HTAB 9: PRINT "*";
4930  NEXT J
4940  IF  PEEK ( - 16384) > 127 THEN 49
60
4950  NEXT I
4960  POKE  - 16368,0
4970  PRINT  CHR$ (7)
4980  RETURN
4990  REM  **
5000  REM
5010 Z$ = "104168104166223154072152072096"
5020  FOR I = 1 TO 10
5030  POKE 767 + I, VAL ( MID$ (Z$,I *
3 - 2,3))
5040  NEXT
5050  RETURN
```

Questo programma è disponibile su disco. Per l'ordinazione utilizzare il coupon "Servizio Software" a pagina 130.

Simulazione e Pocket

Analisi di un problema gestionale mediante un metodo di simulazione con la Casio FX-702P

(1) Offerta (giorni feriali)	0	1	2	3	4	5	6	7
(2) Frequenze	3	9	15	20	22	14	10	7
(3) Probabilità	0.03	0.09	0.15	0.20	0.22	0.14	0.10	0.07
(4) Probabilità cumulata	0.03	0.12	0.27	0.47	0.69	0.83	0.93	1

Figura 2 - Offerta di guide straordinarie nei giorni feriali (frequenze ottenute considerando 100 giorni feriali nel periodo della stagione turistica).

(1) Offerta (domenica)	0	1	2	3	4	5	6	7
(2) Frequenza	7	13	14	9	4	2	1	0
(3) Probabilità	0.14	0.26	0.28	0.18	0.08	0.04	0.02	0
(4) Probabilità cumulatá	0.14	0.40	0.68	0.86	0.94	0.98	1	1

Figura 3 - Offerta di guide straordinarie di domenica (frequenze ottenute considerando 50 domeniche durante l'anno; la domenica l'offerta non è stagionale).

di G. De Nicolao

Uno degli usi più interessanti e più utili che probabilmente si possano fare di un calcolatore consiste nel servirsene per la soluzione di problemi concreti mediante processi di simulazione. Vi sono infatti problemi di svariato genere (e non solo gestionali), che per la loro complessità o non permettono del tutto una trattazione e risoluzione teorica, oppure la rendono quanto mai lunga e laboriosa. Qualora si tratti di ottimizzare dei parametri (all'interno per esempio di una struttura economica), si potrà procedere attraverso numerosi esperimenti pratici, fino a rendersi conto di quali valori garantiscono il miglior funzionamento possibile del processo in esame. Tuttavia queste verifiche pratiche possono richiedere molto tempo e risultare dispendiose in proporzione ai risultati ottenibili. L'uso del calcolatore, che mediante un modello matematico simula il funzionamento del sistema, permette di eseguire in tempo ridotto esperimenti che avrebbero richiesto tempi di realizzazione dell'ordine degli anni con conseguenti enormi spese. Naturalmente il problema si sposta sulla possibilità di ottenere un modello matematico della realtà in esame e sull'affidabilità di tale modello in relazione agli scopi prefissati. L'analisi presentata per l'utilizzazione della Casio FX-702P permette, in circa 1,5 ore di elaborazione (!), di risolvere una gestione di agenzia turistica, ma il metodo è ovviamente applicabile ad altre situazioni che il lettore vorrà studiare.

Il problema

Consideriamo un'agenzia turistica che organizza ogni giorno delle gite turistiche. In relazione alla domanda turistica (variabile nel tempo) l'agenzia deve assumere delle guide-interpreti. L'agenzia dispone di un certo numero *N* di guide-interpreti stabili che le costano L. 116.480 al giorno. Quando la domanda turistica di guide supera il numero *N*, l'agenzia assume giornalmente delle guide-interpreti supplementari, il cui costo giornaliero è di L. 156.800. Tuttavia l'offerta di guide straordinarie non è nè illimitata nè fissa, ma varia di giorno

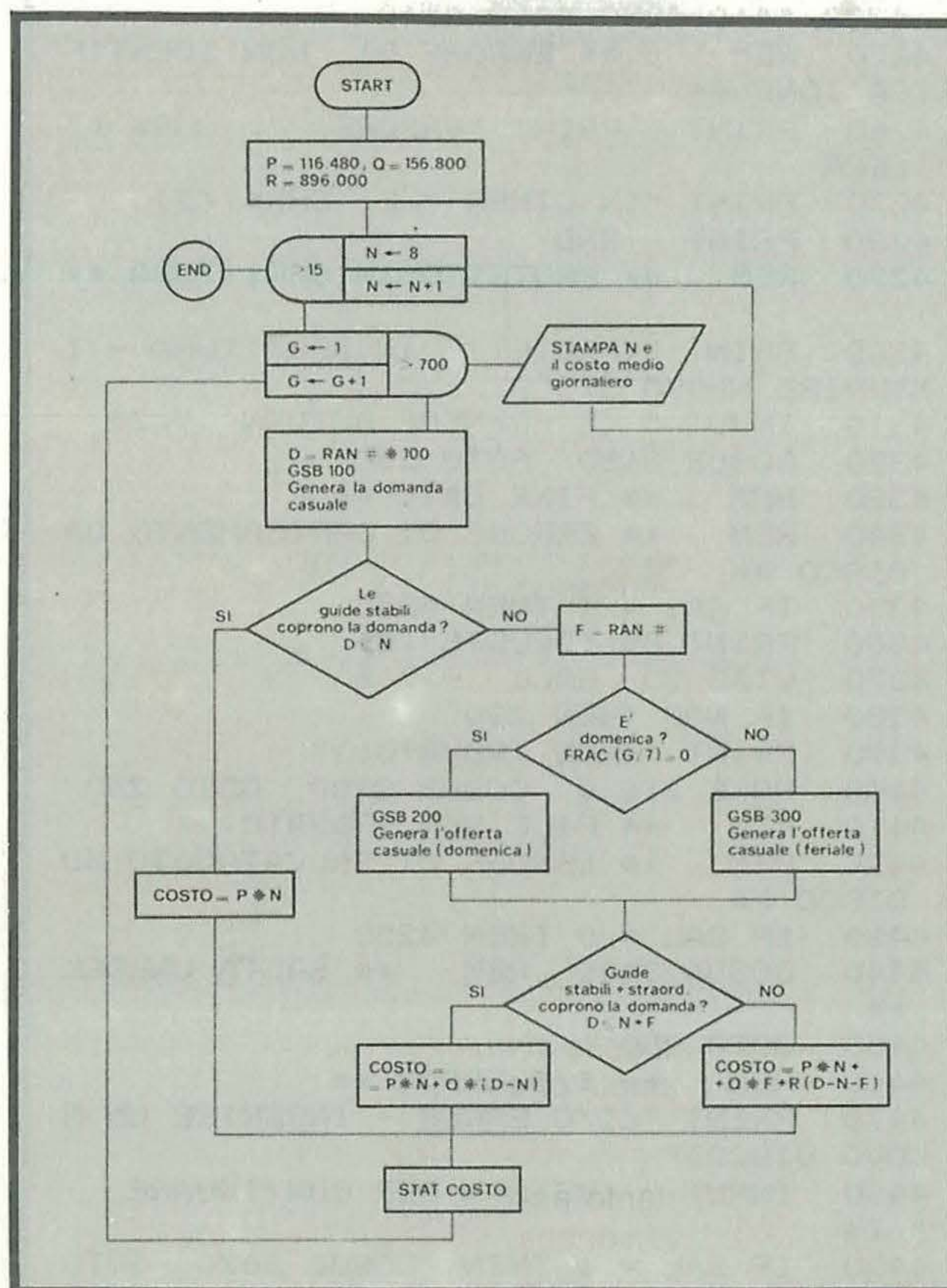



Figura 7 - Diagramma di flusso del programma; P: costo giornaliero di una guida stabile; Q: costo giornaliero di una guida straordinaria; R: mancato guadagno per ogni gita soppressa; N: numero di guide stabili; G: giorno (se è multiplo di 7 è una domenica); D: domanda casuale di guide; F: offerta casuale di guide straordinarie.

	4	5	6	7	8	9	10	11	12	13	14	15	16
(1) Richiesta	4	5	6	7	8	9	10	11	12	13	14	15	16
(2) Frequenza	2	4	4	8	10	12	12	12	10	10	8	6	12
(3) Probabilità	0.02	0.04	0.04	0.08	0.10	0.12	0.012	0.10	0.10	0.10	0.08	0.06	0.02
(4) Probabilita cumulata	0.02	0.06	0.010	0.18	0.28	0.40	0.52	0.64	0.74	0.84	0.92	0.98	1

Figura 1 - Domanda giornaliera di guide-interpreti (frequenze ottenute considerando 100 giorni nel periodo della stagione turistica).

Numero aleatorio equiprobabile				Domande di guide
da	00	a	01	4
da	02	a	05	5
da	06	a	09	6
da	10	a	17	7
da	18	a	27	8
da	28	a	39	9
da	40	a	51	10
da	52	a	63	11
da	64	a	73	12
da	74	a	83	13
da	84	a	91	14
da	92	a	97	15
da	98	a	99	16

Figura 4 - Determinazione casuale delle domande.

Numero aleatorio equiprobabile				Offerta di straordinari
da	00	a	13	0
da	14	a	39	1
da	40	a	67	2
da	68	a	85	3
da	86	a	93	4
da	94	a	97	5
da	98	a	99	6

Figura 6 - Determinazione casuale delle offerte (domenica).

Numero di guide stabili	Costo medio giornaliero
8	1.991.232
9	1.801.120
10	1.603.680
11	1.556.640
12	1.528.096
13	1.580.448
14	1.652.864
15	1.751.008

Figura 8 - Risultati delle simulazioni: si può vedere che il numero di guide stabili che minimizza i costi è 12.

Numero aleatorio equiprobabile				Offerta di straordinari
da	00	a	02	0
da	03	a	11	1
da	12	a	26	2
da	27	a	46	3
da	47	a	68	4
da	69	a	82	5
da	83	a	92	6
da	93	a	99	7

Figura 5 - Determinazione casuale delle offerte (giorni feriali).

```
 10 P=116400:Q=1568
    00:R=896000
 20 FOR N=13 TO 15
 22 SAC
 24 FOR G=1 TO 700
 26 D=RAN#*100:GSB
    100
 28 IF D≤N;C=P*N:GO
    TO 80
 30 F=RAN#*100
 32 IF FRAC (G/7)=0
    ;GSB 200:GOTO 3
    6
 34 GSB 300
 36 IF D≤(N+F);C=P*
    N+Q*(D-N):GOTO
    80
 38 C=P*N+Q*F+R*(D-
    N-F)
 80 STAT C
 82 NEXT G
 84 PRT "N=";N;" MX
    =";MX;"  ";
 86 NEXT N
 90 END
100 IF D≤1;D=4:RET
101 IF D≤5;D=5:RET
102 IF D≤9;D=6:RET
103 IF D≤17;D=7:RET

104 IF D≤27;D=8:RET

105 IF D≤39;D=9:RET

106 IF D≤51;D=10:RE
    T
107 IF D≤63;D=11:RE
    T
108 IF D≤73;D=12:RE
    T
109 IF D≤83;D=13:RE
    T
110 IF D≤91;D=14:RE
    T
111 IF D≤97;D=15:RE
    T
112 D=16:RET
200 IF F≤13;F=0:RET

201 IF F≤39;F=1:RET

202 IF F≤67;F=2:RET

203 IF F≤85;F=3:RET

204 IF F≤93;F=4:RET

205 IF F≤97;F=5:RET

207 F=6:RET
300 IF F≤2;F=0:RET
301 IF F≤11;F=1:RET

302 IF F≤26;F=2:RET

303 IF F≤46;F=3:RET

304 IF F≤68;F=4:RET

305 IF F≤82;F=5:RET

306 IF F≤92;F=6:RET

307 F=7:RET
```

in giorno: può pertanto accadere che l'agenzia debba sopprimere delle escursioni perchè non riesce a coprire la domanda D di guide mediante la somma $N+F$ di guide stabili e guide straordinarie disponibili. In tal caso si calcola una perdita (mancato guadagno) di L. 896.000 per ogni gita soppressa.

Si chiede di determinare il numero N di guide stabili che minimizza i costi medi dell'agenzia turistica.

La soluzione

Questo problema è relativamente semplice (dopotutto si tratta di ottimizzare un solo parametro) e potrebbe essere risolto anche per via analitica, ma desideriamo servircene per esporre un metodo di simulazione. Naturalmente non sarebbe possibile per l'agenzia provare a lavorare per un certo periodo con 8 guide stabili, poi con 9, 10 e così via fino a trovare sperimentalmente quale sia il numero di guide che minimizza i costi. Infatti

questo approccio richiederebbe parecchi anni per ottenere dei costi medi attendibili (data la variabilità casuale della domanda turistica e dell'offerta di guide straordinarie) e perchè l'agenzia non potrebbe permettersi di lavorare per un lungo periodo in condizioni di scarsa competitività.
Il problema può essere risolto mediante la conoscenza delle statistiche relative alla richiesta giornaliera di guide e all'offerta giornaliera di guide straordinarie (si suppone che relativamente all'offerta siano necessarie delle statistiche diverse per i giorni feriali e per le domeniche). Le statistiche sono contenute nelle figure 1, 2, 3.
Per effettuare la simulazione (ovvero per vedere cosa succede all'agenzia in termini di costi quando funziona con un certo numero di guide fisse invece che con un altro) basterà riuscire ad avere una sequenza di domande di guide, e di offerte di guide straordinarie, la cui legge di probabilità sia conforme a quella delle statistiche note. Poi calcolare i costi relativi a queste domande ed offerte. Consideriamo allora la probabilità cumulativa (riga 4 delle figure 1, 2, 3) e facciamo corrispondere: una domanda di 4 unità ad ogni numero 00 o 01; una domanda di 5 unità ad ogni numero dell'intervallo da 02 a 05 (estremi compresi); una domanda di 6 unità ad ogni numero compreso tra 06 e 09 (estremi compresi); ecc.
Otteniamo in questo modo le figure 4, 5, 6. Per ottenere una sequenza di *n* domande casuali basterà ora generare mediante l'istruzione *RAN#* n numeri casuali (compresi tra 0 e 1 esclusi) moltiplicare ciascun numero casuale per 100 e trovare nella figura 4 la domanda corrispondente a ciascun numero. Per quel che riguarda la domanda, siccome le statistiche sono diverse per i giorni feriali e le domeniche, ogni sette numeri bisognerà rivolgersi alla figura 6 invece che alla figura 5.
Dalle statistiche della figura 1 risulta facilmente che il numero *N* di guide stabili che minimizza i costi deve essere compreso tra 8 e 15. Se per ciascuno di questi numeri calcoliamo i costi medi relativi ad una successione casuale (ma che rispetta la legge di probabilità data) di 700 domande ed offerte, sarà come se noi avessimo fatto funzionare l'agenzia per 2 anni con 8 guide, poi altri 2 anni con 9 guide e così via.
Il diagramma di flusso del programma che genera le domande e le offerte e calcola i costi medi è contenuto nella figura 7.
Nella figura 8 sono contenuti i risultati della simulazione che permettono di dire che il numero di guide stabili che minimizza i costi medi è 12. Naturalmente aumentando il numero di giorni (cioè la lunghezza delle sequenze di numeri casuali) lungo i quali si compie la simulazione, la simulazione stessa diviene più affidabile.

Conclusioni

Il problema in esame è piuttosto semplice e l'uso di simulazioni effettuate mediante il calcolatore diviene effettivamente indispensabile solo per casi molto più complessi non altrimenti risolvibili. Lo scopo delle procedure presentata è comunque solo quello di indicare le interessanti potenzialità di simulazione che si possono ottenere anche con un pocket computer di ridottissime dimensioni quale la Casio FX-702P.
(Il problema e l'algoritmo di risoluzione sono tratti da "Manuale di Matematica" di Faure, Kaufmann, Denis - Papin. Casa Editrice ISEDI pp. 703-709).

CBM

Algol

BASIC CBM per simulare battaglie di astronavi complete dei suoni di raggi laser

di E. Bellone

Algol è un programma in BASIC CBM che simula un combattimento spaziale condito con diversi effetti grafici e sonori.
La complessità delle situazioni che si creano nel corso del gioco comporta l'impiego di strategie relativamente contorte, che mettono alla prova la capacità di sintesi e la memoria dell'utilizzatore.
Algol occupa 8195 byte di RAM, variabili a parte, ma omettendo le linee *400-900 2100-2120* (istruzioni) e tutte le *REM*, diventa compatibile con le versioni più economiche del PET.
Gli infelici privi di un'interfaccia audio possono tranquillamente fare a meno di impostare le linee *1020, 5000, 5010, 9040, 9510, 9520, 9530, 9540, 10530* e *11600.*

Descrizione del gioco

Il programma inizialmente colloca 2 navi ammiraglie da 4 caselle, 4 ricognitori da una casella e 10 zone meteoritiche in un settore di 8x8 caselle.

100-120	*Subroutine di attesa, utilizzata dopo aver definito WL (Wait Loop).*
200-250	*Subroutine del generatore nemico: pone il contenuto di S$ in una casella random ancora disponibile in C$. In X e Y viene immesso un valore casuale compreso tra 1 e 8. C$ è una matrice che corrisponde in dimensioni e contenuto al settore di gioco.*
300-320	*Subroutine del radar. Poichè le navi ammiraglie sono rappresentate in C$ con 2 caratteri (+1 = nave n. 1, +2 = nave n. 2), il test della linea 310 ne impedisce la comparsa sul radar. Sono anche di conseguenza ignorati i simboli di posizione precedente ricognitori.*
400-700	*Subroutine istruzioni. Frasi DATA che ne costituiscono il testo.*
800-900	*Stampa delle istruzioni carattere per carattere. Se il carattere è diverso dallo spazio, compare sul video accompagnato da una nota la cui frequenza dipende da K. Sostituendo nella linea 810 K=104 con K=ASC(K$), la nota emessa ha un valore che cambia a seconda del carattere stampato.*
1010	*Clear video e POKE per stampa veloce.*
1020	*POKE per predisposizione uscita audio.*
1030	*Definizione variabili caratteristiche:* *A1 e A2 = numero caselle occupate dalle 2 navi* *R = numero ricognitori* *M = numero zone meteoritiche* *ER = energia residua*
1040	*Z$ posta uguale a "spazio", carattere nullo in questo gioco. La matrice C$, che è sfruttata normalmente entro le dimensioni 1-8/1-8, è riempita di blank con ridondanza per necessità inerenti allo spostamento ricognitori.*
2000-2120	*Presentazione: disegno di una cornice con la parola Algol al centro. La cornice è formata da un solo carattere, sostituibile intervenendo sulla linea 2010 K$="+". Attesa ed eventuale stampa istruzioni.*
3010	*Clear video, ripristino set maiuscolo-/grafico. S$ è posta uguale al simbolo delle due navi ammiraglie (+1, +2) ed il programma salta alla subroutine generatore nemici.*
3020	*Scelta casuale di una direzione verso la quale sistemare le altre 3 caselle delle ammiraglie.*
3030	*Indirizzamento alle 4 zone di sviluppo tramite J.*
3100	*Sviluppo verticale positivo.*
3110	*Se la coordinata verticale è maggiore di 5, l'aggiunta di 3 caselle al di sopra di S$ provocherebbe lo sconfinamento dell'ammiraglia.*
3120	*Test che assicura la disponibilità delle caselle superiori.*
3130	*Completamento dell'ammiraglia e, al termine del ciclo FORK=1TO2 della linea 3010, salto alla fase successiva.*
3200-3430	*Sviluppi realizzati in modo analogo al precedente.*
4010	*Genera R ricognitori.*
4020	*Genera M zone meteoritiche.*
4510-4520	*Richiesta prime coordinate di tiro (formato lettera-numero). POKE 158,0 elimina la possibilità di errori imputabili a caratteri già presenti nella keyboard queue.*
4530-4540	*Apertura di un ciclo FOR-NEXT per l'input di 2 caratteri della tastiera.*
4550	*Y è posta al valore ASCII di K$. Se K$ è il primo carattere impostato, deve essere alfabetico (campo A-H).*
4560	*In DE$ il programma colloca le 2 coordinate di tiro per uso successivo. Stampa di K$ e di un trattino a seguito del 1° carattere.*
4570	*Chiusura del ciclo. 0 è settata ad un valore variabile da 1 a 8, tradotto dal codice ASCII (A=1, B=2, C=3,...).*
4600-4620	*Richiesta livello di energia tramite un solo tasto.*
5010	*Suono emesso dall'armamento dell'Enterprise, variabile a seconda dell'energia impiegata.*

◀ ***Figura 1** - Le parti principali in cui è diviso il programma.*

Foto 2 - Una comunicazione al termine di una fase intermedia. Il game è ulteriormente animato da "drammatici" effetti sonori.

▼

```
ASTRONAVE AMMIRAGLIA NEL SETTORE DI TIRO
ENERGIA ASSORBITA DAL CAMPO PROTETTIVO
ENTERPRISE LOCALIZZATA E COLPITA
STABILIZZATORE QL/052 DANNEGGIATO
PERDITA DI      UNITA' DI ENERGIA
```

◀ *Foto 1 - La videata cruciale: quella della "strumentazione" della vostra astronave* Enterprise. *Fornisce tra l'altro la situazione del settore*

L'utente è a bordo dell'astronave *Enterprise*, che manovra all'esterno del settore. Si dispone di una scorta di 100.000 unità di energia.

Per aprire il fuoco occorre specificare le coordinate di tiro ed il livello di energia prescelto, considerando che con energia 1.000 si possono distruggere ricognitori, con energia 5.000 navi ammiraglie.

In figura 1 potete vedere un esempio della strumentazione.

L'energia residua è un dato molto importante, perchè il programma si interrompe se essa scende al di sotto del minimo valore indispensabile per distruggere la flotta nemica rimanente. Bisogna ricordare inoltre che il punteggio finale varia a seconda di questo parametro.

L'esito del tiro è riassunto come segue:

- Negativo
- Meteoriti polverizzate
- Ricognitore distrutto
- Energia insufficiente
- Nave ammiraglia n. 1/2 colpita
- Nave ammiraglia n. 1/2 distrutta
- Nave ammiraglia n. 1/2 distrutta (bonus E).

Il Settore è riprodotto con il sistema della battaglia navale, con tanto di lettere, numeri e simboli che sono il promemoria dei tentativi precedenti.

Il Radar mostra ricognitori e meteoriti nelle caselle vicine al punto di fuoco.

Time è un contasecondi che indica il tempo che separa il giocatore dall'aggiornamento della situazione.

I ricognitori si spostano casualmente di una casella in direzione orizzontale o verticale, senza invertire comunque il volo, dopo ogni rilevamento radar.

Per rendere il combattimento più movimentato, allo scadere di 15 secondi di inattività a partire dalla stampa della strumentazione si ha un aggiornamento automatico della stessa, che implica una perdita di 500 unità di energia.

Le navi ammiraglie sono immobili, ma protette da un campo di forza che assorbe onde radar ed energie limitate, per cui, oltre a non essere rilevate dai sistemi di avvistamento, rappresentano un grave rischio per il giocatore imprudente.

Infatti impiegare energia 1.000 dove staziona una di queste navi significa segnalare la propria posizione al nemico provocandone l'immediata reazione.

Ai primi 2 errori si ha una penalizzazione di 6.000 e 12.000 unità di energia rispettivamente. Al terzo errore il combattimento termina in un modo piuttosto dequalificante.
Tuttavia non tutti i byte vengono per nuocere: distruggendo una nave ammiraglia con 4 tiri consecutivi si ottiene un prezioso bonus di 20.000 unità di energia. Il gioco si conclude normalmente quando la flotta nemica è ridotta ad un mucchio di relitti vaganti nello spazio.
Il computer stampa alcuni commenti sullo stato dell'*Enterprise* ed assegna un punteggio dipendente dall'energia residua, dal numero dei tentativi fatti, nonchè dagli errori commessi.

Come si gioca

Dopo aver trascorso un bel po' di tempo per impostare *Algol*, potrete finalmente assaporare i primi risultati del vostro lavoro.
Battete *RUN* e richiedete le istruzioni. La visualizzazione avviene in un modo assai particolare, che ricorda una telescrivente all'opera.
In seguito il computer stabilisce casualmente l'ubicazione della flotta nemica e delle zone meteoritiche.
A questo punto dovrete indicare le prime coordinate di tiro e la quantità di energia da erogare al laser, premendo tre tasti in successione, poichè l'input è semplificato mediante l'istruzione *GET*. Udirete un suono diverso a seconda dei dati impostati.
Per scegliere il valore delle prime coordinate affidatevi alla fortuna, evitando le caselle periferiche per ottimizzare la copertura radar.
È buona norma selezionare energia 1.000 sino alla distruzione dei ricognitori oppure sino ai primi scontri con navi ammiraglie; conviene in pratica rischiare 6.000 o 12.000 unità di energia (prima e seconda penalizzazione per energia insufficiente) piuttosto che dare fondo alle proprie scorte.
Al secondo errore bisogna però essere meno parsimoniosi, per salvaguardare la *Enterprise* da un'ulteriore individuazione che provocherebbe la morte di tutto l'equipaggio.
Il punteggio finale è calcolato anche in base ai dann subiti.
Nel corso del gioco fotografate mentalmente il contenuto del monitor, ed in particolare la posizione delle zone meteoritiche nel radar, per evitare tentativi inutili nella ricerca di ricognitori e ammiraglie, che non possono coesistere nella stessa casella delle meteoriti.
Quando siete prossimi allo scadere dei 15 secondi fatidici, ed avete recepito i dati che vi interessano, premete 'spazio' per introdurre le nuove variabili di tiro.
Il programma può ora aspettare indefinitivamente l'elaborazione (umana) dei dati della strumentazione. Condizionatevi a separare la fase lettura dalla fase cerebrale vera e propria.
Quando riuscite ad individuare un nuovo ricognitore sul radar, è opportuno determinare la traiettoria aspettando il successivo scanning. Siete così in grado di eliminare una possibile mossa del mezzo (quella precedente) e, considerando altre limitazioni al suo spostamento (margini del settore, meteoriti), potrete distruggerlo.
Ricognitori dal volo troppo rettilineo possono provocare ragionevoli interrogativi sulla presenza di ammiraglie.
Una piccola nota. Il programma produce lo stesso suono per qualsiasi tiro con esito positivo, quindi aspettate a cantar vittoria al momento dell'esplosione: potrebbe trattarsi di un'irrilevante zona meteoritica.

```
10 REM**********************************
15 REM*           A L G O L            *
20 REM*                                *
25 REM*           BIT  1983            *
30 REM**********************************
50 GOTO1000
100 REM"▮▮▮▮▼▼ SUB PAUSA
110 FORW=0TOWL:NEXTW
120 RETURN
200 REM"▮▮▮▮▼▼ SUB GEN NEMICI
210 X=INT(8*RND(1)+1):IFX>8THEN210
220 Y=INT(8*RND(1)+1):IFY>8THEN220
230 IFC$(X,Y)<>Z$THEN210
240 C$(X,Y)=S$
250 RETURN
300 REM"▮▮▮▮▼▼ SUB RADAR
310 IFLEN(S$)=2THENS$=Z$
320 RETURN
400 REM"▮▮▮▮▼▼ SUB ISTRUZIONI
410 DATA"▮ISTRUZIONI ▮ ♥IETE AL COMANDO DELL'▮NTERP"
420 DATA"RISE, IN UN SETTORE AMPIO 8X8 CASELLE, D"
430 DATA"OVE VI SONO 10 ZONE METEORITICHE (▮). LA"
440 DATA"FLOTTA NEMICA E' FORMATA DA 2 NAVI AMMIR"
450 DATA"AGLIE DI 4 CASELLE (▮▮) IMMOBILI E 4 RICO"
460 DATA"GNITORI DI UNA CASELLA (▮▮) CHE SI SPOSTANO IN"
470 DATA" SENSO ORIZZONTALE O VERTICALE, SENZA"
480 DATA" INVERTIRE COMUNQUE IL VERSO DEL VOLO."
490 DATA"\ETEORITI, RICOGNITORI ED AMMIRAGLIE NON"
500 DATA"POSSONO COESISTERE NELLA STESSA CASELLA."
510 DATA"♠D OGNI TIRO BISOGNA SPECIFICARE LE COOR"
520 DATA"DINATE (♠-I,1-8) ED IL LIVELLO DI ENERG"
530 DATA"IA (1000 PER DISTRUGGERE ▮▮, 5000 PER DA"
540 DATA"NNEGGIARE ▮▮). LE NAVI ▮▮ SONO PROTETTE DA"
550 DATA" UN CAMPO ELETTRICO CHE ASSORBE ONDE RADAR"
560 DATA"ED ENERGIE LIMITATE, PERCIO' IMPIEGARE E"
570 DATA"NERGIA 1000 IN UNA CASELLA DOVE STAZIONA"
580 DATA"UNA ▮▮ SIGNIFICA FAR RILEVARE E COLPIRE L"
590 DATA"A ▮NTERPRISE. ▮L BONUS ENERGY E' ASSEGNA"
600 DATA"TO QUANDO UNA ▮▮ E' DISTRUTTA CON 4 TIRI "
610 DATA"CONSECUTIVI. ▮ISPONETE DI UNA SERIE DI S"
620 DATA"TRUMENTI TRA I QUALI UN QUADRO RIASSUNTI"
630 DATA"VO ED UN RADAR CHE RILEVA ▮▮ E LA POSIZIO"
640 DATA"NE PRECEDENTE DEI ▮▮. LA STRUMENTAZIONE "
650 DATA"VIENE AGGIORNATA AD OGNI AZIONE, OPPURE D"
660 DATA"OPO 15 SECONDI DI STASI CON UN DECREMENTO D"
670 DATA"I 500 UNITA' DI ENERGIA. ▮L PUNTEGGIO"
680 DATA"FINALE E' STABILITO IN BASE AL NUMERO DE"
690 DATA"I TENTATIVI ED ALL'ENERGIA RESIDUA.     "
700 DATA"▮IUONA FORTUNA!"
800 PRINT"▮▮":POKE59468,14:FORX=1TO30:READI$:FORY=1TOLEN(I$)
810 K$=MID$(I$,Y,1):PRINTK$;:K=104:IFK$=Z$THENK=0
820 POKEP,K:WL=5:GOSUB100:POKEP,0:NEXTY
830 IFX<>20THENNEXT:GOTO870
840 PRINT"▮▮REMERE SPAZIO
850 GETK$:IFK$<>Z$THEN850
860 PRINT"▮▮":NEXT
870 PRINTCHR$(13)"▮▮' (S/N)?
880 GETK$:IFK$<>"S"ANDK$<>"N"THEN880
890 IFK$="N"THEN800
900 RETURN
1000 REM"▮▮▮▮▼▼ INIZIALIZZAZIONE
1010 PRINT"▮":POKE59458,62
1020 POKE59467,16:POKE59466,15:P=59464:POKEP,0
1030 A1=4:A2=4:R=4:M=10:ER=100000
1040 Z$=" ":FORX=0TO9:FORY=0TO9:C$(X,Y)=Z$:NEXTY,X
2000 REM"▮▮▮▮▼▼ PRESENTAZIONE
2010 K$="+":FORX=1TO40:S$=S$+K$:NEXT
2020 FORX=1TO19:PRINT"▮"TAB(20-X)LEFT$(S$,2*X):NEXT
2030 FORX=1TO22:PRINTK$SPC(38)K$;:NEXT:PRINT
2040 FORX=1TO19:PRINT"▮"SPC(X)K$SPC(38-X*2)K$:NEXT
2050 PRINT"▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮*** ALGOL ***
2060 WL=1399:GOSUB100
2100 PRINT"▮▮▮ISTRUZIONI (S/N)?
2110 GETK$:IFK$=""THEN2110
2120 IFK$="S"THENGOSUB400
3000 REM"▮▮▮▮▼▼ GEN NEMICI 4 CAS
3010 PRINT"▮":POKE59468,12:FORK=1TO2:S$="*"+RIGHT$(STR$(K),1):GOSUB200
3020 J=INT(4*RND(1)+1):IFJ>4THEN3020
3030 ONJGOTO3100,3200,3300,3400
3100 REM VER+
3110 IFY>5THEN3020
3120 IFC$(X,Y+1)<>Z$ORC$(X,Y+2)<>Z$ORC$(X,Y+3)<>Z$THEN3020
3130 FORH=Y+1TOY+3:C$(X,H)=S$:NEXTH,K:GOTO4000
3200 REM VER-
3210 IFY<4THEN3020
3220 IFC$(X,Y-1)<>Z$ORC$(X,Y-2)<>Z$ORC$(X,Y-3)<>Z$THEN3020
3230 FORH=Y-3TOY-1:C$(X,H)=S$:NEXTH,K:GOTO4000
3300 REM ORIZ+
3310 IFX>5THEN3020
3320 IFC$(X+1,Y)<>Z$ORC$(X+2,Y)<>Z$ORC$(X+3,Y)<>Z$THEN3020
3330 FORH=X+1TOX+3:C$(H,Y)=S$:NEXTH,K:GOTO4000
3400 REM ORIZ-
3410 IFX<4THEN3020
3420 IFC$(X-1,Y)<>Z$ORC$(X-2,Y)<>Z$ORC$(X-3,Y)<>Z$THEN3020
3430 FORH=X-3TOX-1:C$(H,Y)=S$:NEXTH,K
4000 REM"▮▮▮▮▼▼ GEN NEMICI 1 CAS
4010 FORK=1TOR:S$="┴":GOSUB200:NEXTK
4020 FORK=1TOM:S$="▮":GOSUB200:NEXTK
4500 REM"▮▮▮▮▼▼ ING DATI
4510 PRINT"▮HO STABILITO LA POSIZIONE DEI NEMICI.
4520 PRINT"▮COORDINATE DI TIRO (A-H 1-8)? ";:POKE158,0
4530 FORX=1TO2
4540 GETK$:IFK$=""THEN4540
4550 Y=ASC(K$):IFX=1ANDY<65ORY>72THEN4540
4560 DE$(X)=K$:PRINTK$;:IFX=1THENPRINT"-";
4570 NEXT:PRINT:O=ASC(DE$(1))-64:V=VAL(DE$(2))
4600 PRINT"▮ENERGIA (▮1▮▮▮▮000 O ▮5▮▮▮▮000)?";
4610 GETK$:IFK$<>"1"ANDK$<>"5"THEN4610
4620 ET=VAL(K$)*1000:PRINTET
5000 REM"▮▮▮▮▼▼ SUONO ENERGIA
5010 FORX=1TO10:FORY=1TO8:POKEP,(150-Y*ET/1000):NEXTY,X:POKEP,0
6000 REM"▮▮▮▮▼▼ TEST TIRO
6010 TEN=TEN+1:ER=ER-ET:CDT$=C$(O,V)
6020 IFLEFT$(CDT$,1)="*"ANDET=1000THENFE=1:GOTO6100
6030 C$(O,V)=Z$:PT$(O,V)=Z$:K$=LEFT$(CDT$,1):IFK$<>"P"THENPT$(O,V)=K$
6040 IFLEFT$(CDT$,1)="*"ANDET=5000THENESI$="NAVE AMMIRAGLIA N.":GOTO6500
6050 IFCDT$="┴"THENESI$="RICOGNITORE DISTRUTTO":R=R-1:GOTO6100
6060 IFCDT$="▮"THENESI$="METEORITI POLVERIZZATE":M=M-1:GOTO6100
6070 ESI$="NEGATIVO"
6100 HB=0
6110 IFA1+A2+R=0OR((A1+A2)*5000+R*1000)>ERTHENFF=1
6120 GOTO7000
6500 REM"▮▮▮▮▼▼ AMMIRAGLIA COLPITA
6510 IFHB=0THENHB=TEN
6520 S$=RIGHT$(CDT$,1):ESI$=ESI$+S$
6530 IFS$="1"THENA1=A1-1:GOTO6550
6540 A2=A2-1
6550 FORX=1TO8:FORY=1TO8
6560 IFC$(X,Y)="*"+S$THENESI$=ESI$+" COLPITA":GOTO6110
6570 NEXTY,X:ESI$=ESI$+" DISTRUTTA"
6580 IFTEN-HB=3THENESI$=ESI$+" (BONUS E)":FB=1:ER=ER+20000
6590 GOTO6100
7000 REM"▮▮▮▮▼▼ RADAR
```

Figura 2 - Il listato BASIC.

```
7010 S$=C$(O-1,V+1):GOSUB300:S1$=S$:S$=C$(O,V+1):GOSUB300:S1$=S1$+S$
7020 S$=C$(O+1,V+1):GOSUB300:S1$=S1$+S$
7030 S$=C$(O-1,V):GOSUB300:S2$=S$:S2$=S2$+"■ ◘"
7040 S$=C$(O+1,V):GOSUB300:S2$=S2$+S$
7050 S$=C$(O-1,V-1):GOSUB300:S3$=S$:S$=C$(O,V-1):GOSUB300:S3$=S3$+S$
7060 S$=C$(O+1,V-1):GOSUB300:S3$=S3$+S$
7500 REM"▮▮▮▮▼▼ SPOSTAMENTO RICOGNITORI
7510 S$="P2"
7520 IFTEN/2-INT(TEN/2)<>0THENS$="P1"
7530 FORX=1TO8:FORY=1TO8:IFC$(X,Y)="┴"THEN8000
7540 NEXTY,X:GOTO8500
7550 J=INT(4*RND(1)+1):IFJ>4THEN7550
7560 ONJGOTO7600,7700,7800,7900
7600 REM VER+
7610 IFY+1>8ORC$(X,Y+1)<>Z$THEN7550
7620 C$(X,Y+1)="R":GOTO7540
7700 REM VER-
7710 IFY-1<1ORC$(X,Y-1)<>Z$THEN7550
7720 C$(X,Y-1)="R":GOTO7540
7800 REM ORIZ+
7810 IFX+1>8ORC$(X+1,Y)<>Z$THEN7550
7820 C$(X+1,Y)="R":GOTO7540
7900 REM ORIZ-
7910 IFX-1<1ORC$(X-1,Y)<>Z$THEN7550
7920 C$(X-1,Y)="R":GOTO7540
8000 REM"▮▮▮▮▼▼ TEST RICOGNITORE BLOCCATO
8010 K=0
8020 IFC$(X,Y+1)=Z$ANDY<8THENK=K+1
8030 IFC$(X,Y-1)=Z$ANDY>1THENK=K+1
8040 IFC$(X+1,Y)=Z$ANDX<8THENK=K+1
8050 IFC$(X-1,Y)=Z$ANDX>1THENK=K+1
8060 IFK>0THENC$(X,Y)=S$:GOTO7550
8070 GOTO7540
8500 REM"▮▮▮▮▼▼ CLEAR P1/P2 E SOST R
8510 IFS$="P1"THENS$="P2":GOTO8530
8520 S$="P1"
8530 FORX=1TO8:FORY=1TO8
8540 IFC$(X,Y)=S$THENC$(X,Y)=Z$
8550 IFC$(X,Y)="R"THENC$(X,Y)="┴"
8560 NEXTY,X:IFFE=1THENFE=0:GOTO9000
8570 IFFR=1THENRETURN
8580 GOTO9500
9000 REM"▮▮▮▮▼▼ ENERGIA INSUFFICIENTE
9010 PRINT"▯":ESI$="ENERGIA INSUFFICIENTE":C0=C0+1:Y=C0*6000:ER=ER-Y
9020 FORX=0TO99
9030 POKE450*RND(1)*RND(1)+33268-450*RND(1)*RND(1),160
9040 POKEP,43-X/3-10*RND(1)
9050 NEXT:POKEP,0:GOSUB100
9100 PRINT"▯▯▯ASTRONAVE AMMIRAGLIA NEL SETTORE DI TIRO
9110 PRINT"ENERGIA ASSORBITA DAL CAMPO PROTETTIVO
9120 PRINT"▯ENTERPRISE LOCALIZZATA E ";
9130 IFC0=3THEN9300
9140 PRINT"COLPITA
9150 IFC0=2THENPRINT"▯GRUPPO PROPULSORI DSX/9 IN AVARIA
9160 IFC0=1THENPRINT"▯STABILIZZATORE QL/052 DANNEGGIATO
9170 PRINT"▯PERDITA DI"Y"UNITA' DI ENERGIA
9180 S$="":IFY>6000THENS$="_▮▮
9190 FORX=Y-700TOY:PRINT"▯▮▮▮▮▮▮▮▮▮▮▮▮";
9200 IFX=YTHENPRINT"▯____"S$"▮▮▮▮▮▮";
9210 PRINTMID$(STR$(X),2):NEXT:WL=999:GOSUB100:GOTO10000
9300 PRINT"▯__________▮▮▮▮▮▮▮▮▮▮▮▮DISTRUTTA":WL=5999:GOSUB100:FF=1:GOTO10000
9500 REM"▮▮▮▮▼▼ SUONO ESITO POSITIVO
9510 IFESI$="NEGATIVO"THEN10000
9520 FORX=0TO89:POKEP,200:POKEP,253:POKEP,227:IFRND(1)<.3THENPOKEP,0
9530 NEXT:POKEP,0:WL=269:GOSUB100
9540 IFFB=1THENFB=0:WL=49:FORX=0TO37:POKEP,118+COS(X)*24:GOSUB100:NEXTX:POKEP,0
10000 REM"▮▮▮▮▼▼ STRUMENTAZIONE
10010 PRINT"▯";
10020 PRINT"▯COORDINATE▮▮▮▮▮▮▮▮▮▮▮▮▮▮______________▯";
10030 PRINT" "DE$(1)"-"DE$(2)
10040 PRINTTAB(16)"▯ENERGIA IMPIEGATA"CHR$(13)TAB(16)"________________▯"ET
10050 PRINT"▯TENTATIVI▮▮▮▮▮▮▮▮▮▮▮▮______________▯"TEN
10060 PRINTTAB(16)"▯ENERGIA RESIDUA"CHR$(13)TAB(16)"________________▯"ER
10070 PRINT"▯▯▯▯▯▯▯ESITO
10080 FORX=1TO39:PRINT"‾";:NEXT:PRINTCHR$(13)ESI$
10090 PRINT"▯▯SETTORE          SUPERSTITI
10100 PRINT"________         ________________
10200 FORX=1TO8:PRINT"┌┬┬┬┬┬┬┬┐ ":NEXT
10210 PRINT"________▮▮▮▮▮▮▮▮┌┬┬┬┬┬┬┬┐";
10220 FORK=8TO1STEP-1:PRINTTAB(9)MID$(STR$(K),2)CHR$(13)"▯";:FORH=1TO8
10230 IFPT$(H,K)=""THENPRINT"▯";:GOTO10250
10240 PRINT"▯"PT$(H,K);
10250 NEXTH:PRINT:NEXT
10260 PRINT"▯ABCDEFGH
10300 X=0:IFA1>0THENX=1
10310 IFA2>0THENX=X+1
10320 PRINT"┬┬┬┬┬┬┬┬┬┐"TAB(16)"N."X"NAVI AMMIRAGLIE
10330 PRINTTAB(16)"N."R"RICOGNITORI
10400 PRINTTAB(16)"▯RADAR    TIME▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮______   ____
10410 R1$=STR$(V+1):IFV=8THENR1$="  "
10420 R2$=STR$(V):R3$=STR$(V-1):IFV=1THENR3$="  "
10430 PRINTTAB(16)"▯"S1$"■"R1$"▮▮▮▮▮▮▯▯"S2$"■"R2$"▮▮▮▮▮▮▯▯"S3$"■"R3$
10440 R1$=CHR$(O+63):IFO=1THENR1$=" "
10450 R2$=DE$(1):R3$=CHR$(O+65):IFO=8THENR3$=" "
10460 PRINT"▯▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮"R1$+R2$+R3$"▮▮▮▮▮▮┌┬┐▼ SPAZIO
10500 TI$="000000":POKE158,0
10510 GETK$:IFK$=Z$ANDFF=0THENPRINT"▯▯":GOTO4520
10520 PRINT"▯▯▯▯▯▯▯▯▯▯▯▯▯▯▯▯▯▯▯▯"TAB(25)RIGHT$(TI$,2):IFTI$<>"000015"THEN10510
10530 POKEP,100:WL=99:GOSUB100:POKEP,0
10540 FORX=33593TO33601:POKEX,32:NEXT:IFFF=1THEN11000
10550 PRINT"▯";:ER=ER-500:FR=1:GOSUB7000:FR=0:GOTO10020
11000 REM"▮▮▮▮▼▼ MISSIONE ULTIMATA
11010 PRINT"▯▯▯MISSIONE ULTIMATA":FORX=1TO39:PRINT"‾";:NEXT:PRINT"‾
11020 PRINT"▯L'ENTERPRISE ";:K$="HA SUBITO "
11030 ONC0+1GOTO11040,11050,11060,11070
11040 PRINT"NON "K$"DANNI":GOTO11100
11050 PRINTK$"SERI DANNI":GOTO11100
11060 PRINTK$"DANNI GRAVISSIMI":GOTO11100
11070 PRINT"E' STATA DISTRUTTA A CAUSA
11080 PRINT"▯DI UNA IMPRUDENTE STRATEGIA DI GIOCO":GOTO11500
11100 IFA1+A2+R=0THEN11400
11300 PRINT"▯ED E' IN RITIRATA POICHE' NON DISPONE
11310 PRINT"▯DI ENERGIA SUFFICIENTE A FRONTEGGIARE
11320 PRINT"▯IL NEMICO":GOTO11500
11400 X=56000-C0*10000+ER-TEN*1000 :REM MAX 100000 BONUS ESCLUSI
11410 PRINT"▯▯IL TUO PUNTEGGIO E'"X
11500 PRINT"▯▯VUOI GIOCARE ANCORA (S/N)? ";
11510 GETK$:IFK$<>"S"ANDK$<>"N"THEN11510
11520 IFK$="S"THENRUN
11530 PRINTK$:IFX<50000THENPRINT"▯D'ACCORDO, MA DEVI ANCORA ALLENARTI
11600 POKE59458,30:POKE59467,0:POKE59466,0
READY.
```

CBM

PET text editor

Strumento professionale di gestione testi: un PET diventa una efficiente macchina di word processing.

di F. Kalin

Vi ricordate l'articolo apparso sul numero di Dicembre di **Bit** che trattava di un word processing program? In quell'articolo menzionavo il fatto che la routine di ingresso del testo poteva essere modificata a piacere, per renderla più completa ed efficiente: per coloro che non hanno avuto il tempo (o la voglia...) di progettarne una, ecco un text editor fatto su misura e pronto da digitare. Oltre al text editor vero e proprio, è anche presente una parte di programma dedicata alla gestione dell'input-output da memoria di massa, alla visualizzazione o stampa del testo da compilare e ad altre routine di utilità che rendono il programma stesso autosufficiente.
Tra le istruzioni dell'editor, particolare menzione merita il comando *LO*calizza, che consente la ricerca di una parola o di una sua parte in qualunque punto del testo: se ad esempio ad ogni paragrafo viene associato un numero, risulta poi molto facile rintracciarlo per eventuali aggiunte o correzioni. Inoltre la routine di caricamento da file consente di caricare sezioni di testi molto lunghi, e quindi scritti e memorizzati su file diversi e in tempi diversi, una dopo l'altra, ottenendo in memoria un unico testo completo e pronto da impaginare (è cioè simile alle funzioni *MERGE* e *COLLECT* di altri sistemi operativi).

Il programma principale

È quello che gestisce le utilitie, al lancio del programma, appare il simbolo > ed il computer rimane in attesa di uno dei seguenti comandi:
DISPLAY e *PRINT*: permettono il listing, rispettivamente su video o su stampante, del testo contenuto in memoria;
SAVE e *LOAD*: registrazione e caricamento di testi non impaginati (alla richiesta del device, bisogna rispondere 1 per il tape #1, 2 per il tape #2 oppure 8 per l'unità disco);
DELETE: cancella il testo presente in memoria;
FRE: visualizza il numero di byte liberi (come il comando *PRINT FREE (0)* del BASIC);
BYE: termina l'esecuzione del programma;
WPP: salta alla routine di impaginazione;
EDI: entra nell'editor vero e proprio.

I comandi dell'editor

Una volta dato il comando *EDI*, il programma si comporta in modo diverso a seconda che già esista un testo in memoria oppure che la memoria sia libera; in quest'ultimo caso, compare la scritta "[*inserisci il nuovo testo*]" ed è già possibile scrivere senza usare il tasto *RETURN*. Questo tasto è invece indispensabile per inserire i comandi dell'impaginatore (vedere articolo citato e la figura 1): infatti ciascun comando del PWPP deve occupare da solo una linea del testo (ciò non era spiegato bene in quell'articolo: scusate).
A proposito di tasti, ne esistono alcuni che, pur essendo presenti sulla tastiera, non possono essere usati; sono: le due frecce, quella verso l'alto e quella verso sinistra, perchè vengono usate dalla routine di registrazione su memoria di massa al posto dei caratteri "virgola" e "due-punti" (in quanto questi ultimi creano in fase di lettura diversi problemi: vedere a questo proposito la rubrica "*Il ricettario*" su questo stesso numero), i tasti di indirizzamento del cursore escluso il *DELETE* che invece può essere usato normalmente, ed infine il "doppio-apice" che crea problemi sul video.
Per uscire dall'editor e ritornare al programma principale bisogna inserire una linea vuota (ovvero premere due volte il tasto *RETURN*) e alla comparsa dell'asterisco scrivere *EX*: seguirà la scritta "[*uscito*]" e riapparirà il simbolo > (nota: per poter inserire una linea bianca nel testo bisogna, prima del tasto di *RETURN*, scrivere uno spazio, altrimenti si esce dall'editor).
Se invece esiste già un testo in memoria, al comando *EDI* il computer risponde con la scritta "[*NN linee presenti*]" e con la comparsa dell'asterisco: a questo punto, ogni volta che si da *RETURN* il programma visualizza una linea del testo, fino a quando, non essendoci più linee, stampa la scritta "[*EOF*]" (fine del file).
Al posto della linea vuota (cioè del solo tasto *RETURN*), è possibile inserire alcuni comandi; oltre al comando *EX* di cui ho già parlato, gli altri sono:
DE: cancella la riga appena visualizzata e mostra quella successiva;
I: permette l'input di altre linee di testo dopo quella visualizzata (in pratica il funzionamento diviene identico a quando si inserisce un testo nuovo, mentre le linee seguenti vengono spostate per far posto alle nuove);
***NP** nn*: consente di spostarsi in avanti (se nn è positivo) o indietro (nn negativo) all'interno del testo, per saltare gruppi di linee (se nn è troppo negativo appare la scritta "[*TOF*]" - inizio del file -, mentre se è troppo positivo appare "[*EOF*]" già vista);
***AP** stringa*: appende alla linea visualizzata la stringa che segue lo spazio dopo il comando;
C: cambia una parte della linea visualizzata con una nuova stringa; il formato generale è *C*/stringa vecchia/stringa nuova/ (vedere anche gli esempi di figura 2). Il carattere che separa il comando dalle stringhe e le stringhe tra loro può essere un qualunque carattere non alfabetico (deve però essere lo stesso all'interno dell'intero comando); se la stringa vecchia non esiste nella linea visualizzata (ad esempio per un errore di battitura), il programma lo segnala con la scritta "[*Non Trovata*]";
***LO** stringa*: è il comando più potente. Il suo funzionamento ed il suo utilizzo sono già stati spiegati nell'introduzione; qui posso solo aggiungere che il comando effettua la ricerca della stringa a partire dall'ultima linea visualizzata in avanti, in modo che sia possibile usarlo ripetutamente, senza ogni volta dover ripartire dall'inizio del testo. Il programma inoltre visualizza i numeri delle linee scandite e se trova la stringa cercata si ferma visualizzando la linea che la contiene, altrimenti alla fine del testo stampa la scritta "[*Non Trovata*]" e rimane posizionato alla linea alla quale il comando era stato digitato (nel caso sia sfuggito un errore di battitura, questo consente di ripetere il comando senza dover fare altro).

Il caricamento del programma

Differisce un poco se si vuole avere il text editor separato dal PWPP o se si desidera inglobare le due parti in un unico programma (io consiglio quest'ultima possibilità). Nel primo caso il caricamento del listato è tutto ciò che occorre fare: questo però comporta, vista l'assenza dell'impaginatore, la necessità di registrare il testo su file.
Se invece si vuole un programma unico e completo, bisogna avere in memoria il programma PWPP, cancellare le linee da *1* a *91* comprese (vedere articolo citato) e togliere l'istruzione *RETURN* dalla linea *10000*. A questo punto si inserisce il listato del text editor escluse la linea *1000* e la linea *50003*, ed infine si cambia, alla linea *1*, l'istruzione *GOSUB 50000* con *GOSUB*

10000: si ha così a disposizione un unico programma, più facile da usare ma che occupa più memoria (problema, questo della quantità di memoria, ben noto agli utenti dei PET da 8 Kbyte). C'è inoltre da dire che nel programma sono contenuti i menu dei comandi sia del programma principale, sia dell'editor sia dell'impaginatore: il primo è richiamabile digitando *MENU* quando è visualizzato il simbolo > mentre gli altri due digitando sempre *MENU* ma quando si è in *EDI* ed è presente l'asterisco.

Il listato

È diviso essenzialmente in 6 parti: le routine di sistema cioè quelle più usate (linee *4-42*), il programma principale che come tutti i *MAIN* deve essere solo lo smistatore dei lavori, le routine di utilità (linee *140-280*), la routine *EDI* (linee *300-340*) con i suoi comandi (linee *350-710*) ed infine il sottoprogramma di inizializzazione del sistema (linee *50000* e seguenti).
Particolare attenzione va fatta per la linea *60000*: e la routine di stampa chiamata dal comando *PRINT* (vedere linea *140*). È stata messa in tale posizione perchè, se si desidera la stampa di maiuscole e minuscole insieme, risulta necessario usare una routine più complessa e dipendente dal tipo di stampante che si ha a disposizione.
Due parole ora su come comportarsi di fronte ad un'interruzione prematura del programma. Questa può dipendere da due cause: la prima dovuta alla pressione del tasto *RUN/STOP* anzichè quello di *RETURN* (nel modello 3032 sono attigui), la seconda dovuta ad una svista su un comando del PWPP (vedere linea *1220* di quel listato). Nel primo caso basta digitare *CONT*, mentre nel secondo bisogna dare il comando *GOTO 100* che riporta all'interno del programma principale senza perdere il testo presente in memoria (ciò che invece accadrebbe usando il comando *RUN*).
Un'ultima spiegazione per i patiti dei dettagli: perchè le routine di sistema sono in cima al programma? La risposta è semplice: perchè devono essere veloci. Il BASIC infatti quando esegue un'istruzione *GOTO* o *GOSUB* cerca la linea a cui saltare partendo dall'alto, e quindi più in alto sono le routine più presto ci arriva: questa piccola divagazione teorica forse potrà servire ai creatori di giochi, per complicare un po' di più la vita a chi si dovrà cimentare con programmi più veloci.
Il sottoprogramma di gestione dei file è stato progettato per il BASIC livello 3.0 (cioè per i modelli 3000); per i modelli 4000 dotati del BASIC livello 4.0 è necessario modificare nella linea 196 la stringa:

IF L% >0 AND LEN (S$$) >1

con la stringa

IF LEFT$ (S$$,1) =CHR$ (255)

```
...
Fine del paragrafo<CR>
.bp 2<CR>
.ta 3<CR>
Inizio nuovo paragrafo...
```

Figura 1 - Il tasto RETURN *va usato solo per inserire i comandi del PWPP che devono essere da soli su una riga (<(CR)> è il tasto* RETURN).

```
*

porta

*c/p/t/

torta           cambio della p con una t

*c//s/

storta          aggiunta all'inizio della linea

*c.tor.forna.

sfornata        cambio di piu' caratteri

*c/fur/uo/

[non trovata]   esempio di errore (la
                stringa fur non esiste)
*c/for/uo/

suonata

*

Attenzione: se in una linea sono presenti
due stringhe uguali, il programma cambiera'
sempre la prima: ad esempio per passare da
SUONATA a SUONATORI bisogna scrivere

 *c.ta.tori.

come si puo' vedere facendo alcune prove;
infatti se si scrive

 *c.a.ori.

si ottiene la stringa SUONORITA che non e'
certamente il risultato richiesto.
```

Figura 2 - Esempi di uso del comando C.

```
PET TEXT EDITOR

 1 PRINT"▯■■":GOSUB 50000:GOTO 100
 2 REM       *** INIZIO ROUTINES DI S
ISTEMA  ***
 3 REM
 4 PRINT"_■";:FOR K=1 TO 50
 5 GET R$:IF R$<>"" THEN 9
 6 NEXT K:PRINT" ■";:FOR K=1 TO 50
 7 GET R$:IF R$<>"" THEN 9
 8 NEXT K:GOTO 4
 9 PRINT" ■";:K=50:NEXT K:R=ASC(R$):RET
URN
 10 A$=""
 11 GOSUB 4:IF R=13 THEN PRINT:RETURN
 12 IF R<>20 THEN 16
 13 L=LEN(A$):IF L>0 THEN PRINT"■ ■";
 14 IF L<2 THEN 10
 15 A$=LEFT$(A$,L-1):GOTO 11
 16 IF R=32 AND LEN(A$)>60 THEN PRINT:R
ETURN
 19 PRINT R$;:A$=A$+R$:GOTO 11
 20 B$="":LA=LEN(A$):N=1
 21 IF N>LA THEN B$=A$:LB=LEN(B$):GOTO
26
 22 R$=MID$(A$,N,1):R=ASC(R$):IF B=1 AN
D R$=CS$ THEN 25
 23 IF B=0 AND (R<65 OR R>91) THEN CS$=
R$:GOTO 25
 24 B$=B$+R$:N=N+1:GOTO 21
 25 LA=LA-N:LB=LEN(B$):IF LA THEN A$=RI
GHT$(A$,LA):RETURN
 26 LA=0:A$="":RETURN
 30 C=0:FOR I=1 TO CM%(B):IF B$=CM$(B,I
) THEN C=I
 31 NEXT I:RETURN
 35 GOSUB 10:IF A$="" THEN RETURN
```

Figura 3 - Il listato BASIC.

Seguito listato PET text editor

```
 37 L%=L%+1:M%=M%+1:IF L%=M% THEN P$(L%
-1)=A$:GOSUB 700:GOTO 35
 38 FOR I=M% TO L%+1 STEP -1:P$(I)=P$(I
-1):NEXT I:P$(L%)=A$:GOSUB 700:GOTO 35
 40 LP=LEN(S$):C=0:LM=LP-LB+1
 42 FOR I=1 TO LM:IF MID$(S$,I,LB)=B$ T
HEN C=1:P=I:I=LM
 44 NEXT I:RETURN
 50 REM
 60 REM      ***  FINE ROUTINES DI SISTE
MA  ***
 70 REM
 97 REM
 98 REM      ***  PROGRAMMA PRINCIPALE
***
 99 REM
 100 PRINT"■>";:GOSUB 10:IF A$="" THEN
100
 110 GOSUB 20:B=1:GOSUB 30:IF C=0 THEN
PRINT"[COMANDO ILLEGALE]":GOTO 100
 120 IF C<8 THEN ON C GOSUB 140,150,160
,170,180,190,250:GOTO 100
 130 C=C-7:ON C GOTO 300,240,1000
 134 REM
 136 REM      ***  UTILITY ROUTINES   ***
 138 REM
 140 OPEN 4,4:DE=4:IF M% THEN FOR I=0 T
O M%-1:S$=P$(I):GOSUB 60000:NEXT I
 145 CLOSE 4:RETURN
 150 PRINT:IF M%=0 THEN 158
 151 PRINT"■■■ -S- FERMA IL LIST, -Q- L
O RIPRENDE"
 152 GET R$:IF R$<>"Q" THEN 152
 153 FOR I=0 TO M%-1:GET R$:IF R$ <>"S"
 THEN 156
 154 GET R$:IF R$<>"Q" THEN 154
 156 PRINT P$(I):FOR J=1 TO 100:NEXT J,
I
 158 PRINT:RETURN
 160 PRINT"■"FRE(P)"BYTES FREE■":RETURN
 170 FOR I=0 TO M%:P$(I)="":NEXT I:M%=0
:RETURN
 180 IF M%=0 THEN PRINT"■NESSUN FILE DA
SALVARE■":RETURN
 182 SA=1:GOSUB 210:IF ER=1 THEN RETURN
 184 PRINT"■ATTENDERE.":Z$=CHR$(13):XX=
0:FOR I=0 TO M%-1
 186 S$=P$(I):PR$=",←":GOSUB 650:PR$=":
↑":GOSUB 650
 188 PRINT#4,S$;Z$:NEXT I:CLOSE 4:CLOSE
 1:RETURN
 190 SA=0:GOSUB 210:IF ER=1 THEN RETURN
 192 PRINT"■ATTENDERE.":L%=0
 194 INPUT#4,S$:PR$="↑:":GOSUB 650:PR$=
"←,":GOSUB 650
 196 IF L%>0 AND LEN(S$)>1 THEN S$=RIGH
T$(S$,LEN(S$)-1)
 198 P$(M%)=S$:L%=L%+1:M%=M%+1
 200 IF ST<>64 THEN 194
 202 CLOSE 4:CLOSE 1:M%=M%-1:RETURN
 210 PRINT"■NOME DEL FILE: ";:GOSUB 10:
B$=A$:PRINT"■DEVICE: ";:GOSUB 10:ER=0
 212 DE=VAL(A$):IF DE=4 THEN OPEN4,4:RE
TURN
 214 IF DE<>8 THEN OPEN4,DE,SA,B$:RETUR
N
 216 PRINT"■DRIVE: ";:GOSUB 10:S$=A$:A$
="N"
 217 IF SA=1 THEN PRINT"■CANCELLO IL VE
CCHIO FILE [S/N]? ";:GOSUB 10
 218 OPEN 1,8,15:PRINT#1,"I"+S$:IF A$="
S" THEN S$="@"+S$
 220 A$="R":IF SA=1 THEN A$="W"
 222 B$=S$+":"+B$+",S,"+A$:OPEN 4,8,4,B
$
 224 INPUT#1,EN$,EM$,ET$,ES$
 226 IF EN$="00" THEN RETURN
 228 PRINT "■";EM$;" ERROR":CLOSE 4:CLO
SE 1:ER=1:RETURN
 240 PRINT"■■ESECUZIONE TERMINATA.":PRI
NT"■ARRIVEDERCI.":END
 250 PRINT"■■■    MENU■"
 255 PRINT"EDI          INPUT DEL TESTO"
 260 PRINT"DISPLAY   VISUALIZZAZIONE TES
TO"
 265 PRINT"PRINT       STAMPA DEL TESTO"
 270 PRINT"SAVE        REGISTRAZIONE TESTO
"
 275 PRINT"LOAD        CARICAMENTO TESTO"
 280 PRINT"FRE         MEMORIA LIBERA"
 285 PRINT"DELETE      CANCELLAZIONE TESTO
"
 290 PRINT"WPP         CHIAMATA DELL'IMPAG
INATORE"
 295 PRINT"BYE         TERMINE ELABORAZION
E":PRINT:RETURN
 297 REM
 298 REM      ***  ROUTINE EDITOR   ***
 299 REM
 300 PRINT"■■[";:L%=-1:IF M% THEN PRINT
M% "LINEE PRESENTI]":GOTO 310
 305 PRINT"INSERISCI IL NUOVO TESTO]":G
OSUB 35
 310 PRINT"*";:GOSUB 10:IF A$<>"" THEN
335
 315 L%=L%+1
 320 IF L%>=0 AND L%<M% THEN PRINT P$(L
%):GOTO 310
 325 IF L%<0 THEN L%=-1:PRINT"[TOF]":GO
TO 310
 330 IF L%>=M% THEN L%=M%:PRINT"[EOF]":
GOTO 310
 335 B=0:GOSUB 20:B=2:GOSUB 30:IF C=0 T
HEN PRINT"[COMANDO ILLEGALE]":GOTO 310
 340 ON C GOTO 350,360,365,370,375,400,
500,450
 344 REM
 345 REM      ***  ROUTINES COMANDI EDIT
OR***
 346 REM
 350 PRINT"■[USCITO]":GOTO 100
 360 L%=L%+VAL(CS$+A$):GOTO320
 365 GOSUB 35:GOTO 310
 370 FOR I=L% TO M%-1 : P$(I)=P$(I+1):N
EXT I:M%=M%-1:P$(M%)="":GOTO 320
 375 B=1:S$=P$(L%):GOSUB 20:GOSUB 40:IF
 C=0 THEN PRINT"[NON TROVATA]":GOTO 310
 380 GOSUB 20:IF P=1 THEN S$=B$+RIGHT$(
S$,LM-1):GOTO 395
 385 IF LM=P THEN S$=LEFT$(S$,LM-1)+B$:
GOTO 395
 390 S$=LEFT$(S$,P-1)+B$+RIGHT$(S$,LM-P
)
 395 P$(L%)=S$:GOTO 320
 400 P$(L%)=P$(L%)+A$:GOTO 320
 450 B=1:GOSUB 20:FOR J=L%+1 TO M%-1:S$
=P$(J):GOSUB 40
 460 IF C<>0 THEN L%=J:J=M%-1
 470 IF C=0 THEN PRINT"[VISTA LINEA"J"]
"
 480 NEXT J:IF C<>0 THEN 320
 490 PRINT"[NON TROVATA]":GOTO 310
 500 PRINT"■■■ EDITOR (1) O WPP (2) ? "
;:GOSUB 10:IF VAL(A$)=1 THEN 600
 502 PRINT"■■.=C    DEFINIZIONE CARATTER
E CONTROLLO"
 504 PRINT".BP    INIZIO PAGINA":PRINT".
MA    MODIFICA MARGINI"
 506 PRINT".TA    TABULAZIONE INIZIO CAP
OVERSO":PRINT".CE    CENTRATURA TITOLI"
 508 PRINT".IN    INTESTAZIONE PAGINA":P
RINT".RT    NUMERO LINEE PER PAGINA"
```

```
 510 PRINT".LS   SPAZIATURA TRA LE LINE
E":PRINT".SP   STAMPA DI LINEE BIANCHE"
 512 PRINT".NF   INTERRUZIONE DELL'IMPA
GINAZIONE"
 514 PRINT".FI   RIPRESA DELL'IMPAGINAZ
IONE"
 516 PRINT".NE   SALTA A PAGINA NUOVA S
E MANCANO":PRINT"          MENO DI N RIGHE"
 518 PRINT"▒*":GOTO 320
 600 PRINT"▒▒I         INPUT":PRINT"EX     U
SCITA":PRINT"DE     CANCELLAZIONE RIGA"
 610 PRINT"C      CAMBIO DI UNA PARTE DE
LLA RIGA"
 615 PRINT"AP     AGGIUNTA AD UNA RIGA"
 620 PRINT"NP     SPOSTAMENTO DALLA RIGA
LISTATA"
 630 PRINT"LO     CERCA UNA STRINGA IN T
UTTO IL":PRINT"         TESTO"
 640 PRINT"▒*":GOTO 320
 650 DO$=RIGHT$(PR$,1):PR$=LEFT$(PR$,1)
 655 W=0:WW=LEN(S$):FORZ=1TOWW:IFMID$(S
$,Z,1)=PR$THENW=Z:Z=WW
 660 NEXTZ:IF W=0 THENRETURN
 665 IF W=1THENB$=RIGHT$(S$,WW-1):S$=""
:GOTO680
 670 IF W=WW THENB$="":S$=LEFT$(S$,WW-1
):GOTO680
 675 B$=RIGHT$(S$,WW-W):S$=LEFT$(S$,W-1
)
 680 S$=S$+DO$+B$:GOTO655
 700 IF FRE(P)<1000 OR M%>240 THEN PRIN
T" ATTENZIONE: POCA MEMORIA"
 710 RETURN
 1000 PRINT"▒IMPAGINATORE ASSENTE.":GOT
O 100
 50000 POKE59468,14:DIM CM$(2,10),CM%(2
):FOR I=1 TO 2:READ CM%(I):NEXT I
 50003 DIM P$(255)
 50005 FOR I=1 TO 2:FOR J=1 TO CM%(I):R
EAD CM$(I,J):NEXT J,I:M%=0:GOTO 250
 50010 DATA 10,8,PRINT,DISPLAY,FRE,DELE
TE,SAVE,LOAD,MENU,EDI,BYE,WPP
 50020 DATA EX,NP,I,DE,C,AP,MENU,LO
 60000 PRINT#4,S$:RETURN
```

Lista simboli grafici

```
   1 : 1   SHIFT HOME =CHR$(147)
       2   CRSR↑ =CHR$(17)

   4 : 1   SHIFT CRSR← =CHR$(157)

   6 : 1   SHIFT CRSR← =CHR$(157)

   9 : 1   SHIFT CRSR← =CHR$(157)

  13 : 1   SHIFT CRSR← =CHR$(157)
       1   SHIFT CRSR← =CHR$(157)

 100 : 1   CRSR↑ =CHR$(17)

 151 : 1   SHIFT HOME =CHR$(147)
       2   CRSR↑ =CHR$(17)

 160 : 1   CRSR↑ =CHR$(17)
       1   CRSR↑ =CHR$(17)

 180 : 1   CRSR↑ =CHR$(17)
       1   CRSR↑ =CHR$(17)

 184 : 1   CRSR↑ =CHR$(17)

 192 : 1   CRSR↑ =CHR$(17)

 210 : 1   CRSR↑ =CHR$(17)
       1   CRSR↑ =CHR$(17)

 216 : 1   CRSR↑ =CHR$(17)

 217 : 1   CRSR↑ =CHR$(17)

 228 : 1   CRSR↑ =CHR$(17)

 240 : 2   CRSR↑ =CHR$(17)
       1   CRSR↑ =CHR$(17)

 250 : 1   SHIFT HOME =CHR$(147)
       2   CRSR↑ =CHR$(17)
       1   CRSR↑ =CHR$(17)

 300 : 1   SHIFT HOME =CHR$(147)
       1   CRSR↑ =CHR$(17)

 350 : 1   CRSR↑ =CHR$(17)

 500 : 1   SHIFT HOME =CHR$(147)
       2   CRSR↑ =CHR$(17)

 502 : 2   CRSR↑ =CHR$(17)

 518 : 1   CRSR↑ =CHR$(17)

 600 : 2   CRSR↑ =CHR$(17)

 640 : 1   CRSR↑ =CHR$(17)

1000 : 1   CRSR↑ =CHR$(17)
```

Questo programma è disponibile su cassetta o su disco. Per l'ordinazione utilizzare il coupon "Servizio Software" a pagina 130.

CBM

Effetti bancari

Personal business: operazioni bancarie

di L. Di Martino

Richiami ragioneristici

Una delle operazioni più comuni, per chi ha rapporti con una banca, è quella della negoziazione degli effetti. Ricordiamo che "effetto" è un termine tecnico che denota una cambiale, una tratta, una ricevuta bancaria, e via di seguito.
Gli effetti possono essere ceduti ad una banca per diversi motivi e operazioni: a semplice custodia, a garanzia di altre operazioni, per il semplice incasso, ecc. Le due più note e frequenti cessioni sono per lo sconto e per l'anticipo con accredito in conto SBF (Salvo Buon Fine).
Le cambiali hanno un costo di bollo che fa riflettere, il 12%, si ricorre allora alle semplici ricevute bancarie che hanno un costo di bollo inferiore, da un minimo di Lire 200 ad un massimo di Lire 3.500. La conseguenza più ovvia è che l'intero portafoglio effetti circolante è oggi costituito per l'80% circa da ricevute bancarie.
Queste tuttavia non possono essere scontate, ma vengono accreditate in conto al costo delle sole commissioni, ma con valuta media posticipata.
La differenza sostanziale tra un'operazione di sconto effetti ed un accredito SBF sta nel fatto che, con il primo, il cliente paga subito il relativo interesse ed ottiene il netto ricavo in valuta contante, ossia denaro liquido, mentre con l'accredito SBF egli ottiene ugualmente la disponibilita di denaro, ma con la valuta posticipata come da calcolo di valuta media e, non trattandosi di denaro liquido, pagherà successivamente l'interesse sul conto per lo scoperto di valuta, ma in ragione del solo ammontare di utilizzo e limitatamente al tempo intercorso tra la data di utilizzo e quella in cui ha luogo il "buon fine", ossia matura la valuta SBF.
L'accredito di effetti SBF risulta pertanto l'operazione più frequente perchè più conveniente per il cliente, sia a causa del minor costo di bollo che per la miglior elasticità di utilizzo rispetto ad una operazione di sconto effetti.
Ad ogni operazione in banca viene dunque assegnata la relativa valuta come da cartello bancario; per *"valuta"* ricordiamo che è da intendersi il giorno di inizio conteggio interessi (attivi-/passivi) e non corrisponde quasi mai alla data dell'operazione. Essa gioca un ruolo importante in tutte le operazioni bancarie.
Il programma qui illustrato riguarda esclusivamente le due operazioni di cui si è parlato: *SCONTO EFFETTI* e *ACCREDITO EFFETTI SBF* (calcolo della valuta media).
In genere, quando si ricevono dalla banca gli elaborati relativi alle due operazioni, si rimane regolarmente sconcertati, essendo ... diversi da come li avevamo preventivati. Una migliore conoscenza delle procedure operative seguite dalle banche e delle leggi che la governano possono perciò venirci in aiuto, se non altro prevenendo amare sorprese (*oppure perdite di tempo e figuracce agli sportelli bancari!* NdR). *NEGOZIAZIONE EFFETTI IN BANCA* è un programma che serve a tale scopo, offrendo la possibilità di effettuare una autentica e completa negoziazione effetti, sia *preventivamente* che *successivamente* alla effettiva negoziazione in banca e soprattutto

GIOCHI COL PERSONAL? SI, MA CI VOGLION PURE COSE UTILI, OGNI TANTO!

Si fa in giro tanto baccano sull'home computing, però la realtà è ancora alquanto squallida. Il perchè è subito visto: *scarseggia il software applicativo oppure è ancora troppo caro.* In queste condizioni perchè meravigliarsi se l'elaborazione "casereccia" non decolla, oppure a tutt'oggi (e il fenomeno, se ciò può consolarci, si manifesta persino negli States ...) investe una sparuta avanguardia? Molti di costoro, in realtà, dopo essersi lasciati tentare da un acquisto che, non nascondiamocelo, è piuttosto oneroso, provano a produrre da soli del software pratico per l'uso individuale, però in genere i risultati sono scarsi, non solo perchè programmare è difficile (anzi per molti è un divertimento) ma perchè *programmare stanca*, forse persino più del lavoro di cui parlava Cesare Pavese: si richiede infatti una certa fantasia nell'escogitare gli stessi settori applicativi della personal-computeria e, in ogni caso, un mucchio di tempo. Risultato? Ci si consola con i videogame, più o meno "arcadici", anche sotto la spinta dei più giovani che di simili cose sono giustamente appassionati e inoltre, diciamolo pure, riescono a trascinare anche quegli eterni fanciulloni che si celano sotto le più ruvide e vetuste scorze ... insomma si ripete, coi personal, il già noto fenomeno dei genitori che contendono alla figliolanza l'uso del meccano. Solo che qui, ogni tanto, spunta un preciso e pungente rimorso, ripensando a quanto si è speso e a quanto ci si riproponeva di fare, magari per porre ordine nelle familiari, ma non per questo incasinate finanze. Che può fare **Bit** in tale contesto? Lungi da noi ogni intento di rigirare, sadicamente, quel tal coltello in quella certa piaga. Che il gioco sia il sale della vita corrisponde ad una nostra precisa convinzione e non nasce (solo) dal timore di provocar l'ira del già irritabile *genus distributorum* (a sua volta, riteniamo, non scevro dal complesso di non riuscire a produrre, a bassi costi, questo mitico logiciel individuel).
Così noi ci mettiamo una pezza - a vantaggio di tutti, riteniamo - stimolando la produzione e pubblicando lavori pratici come questo di Lucio Di Martino. Ad esser sinceri siamo orrendamente in torto nei suoi riguardi, non avendo trovato spazio per stamparglielo. Pensate che costui ha saputo ripagare tanta iniqua indifferenza non solo con la paziente attesa ma addirittura, ultimamente, con l'invio di un'edizione del suo elaborato *aggiornata* sulla base degli ultimi dettami del cartello bancario, in vigore dal 1° Ottobre 1982.
L'articolo è abbastanza ben documentato, essendo anche preceduto (su nostra indicazione) dai necessari richiami in materia, il che ci sembra ben corrispondente a quel concetto sopra esposto di segnalare le possibili applicazioni, anche quelle cioè cui taluni potrebbero non avere pensato.

Figura 1 - Ecco quanto prevede il vigente cartello bancario per quanto concerne le due operazioni sconto effetti e accredito eff. SBF. Il tutto viene applicato automaticamente dal programma presentato.
Es. di domiciliazione: un effetto negoziato presso la Banca Topi LTD di Milano, ha la clausola pagabile presso la Istsanpaol sempre di Milano; per la banca negoziatrice l'effetto è fuori piazza anche se pagabile nello stesso comune, comporta una spesa effettiva in più in comm. e valuta per l'appoggio dell'incasso e viene recuperato con la valuta superiore di fuori piazza.

CARTELLO BANCARIO

COMMISSIONI E PROVVIGIONI

- SPESE FISSE, per presentazione .... L. 500 =
- DIRITTO DI BREVITA', per effetto .... L. 3.000 =
- SCONTO EFFETTI
- EFFETTI SBF } per effetto .... L. 2.500 =
- MINIMO DI SCONTO, per effetto ....L. 2.000 =
- PROVVIGIONE DI RINNOVO, 1/4, ossia 0,25% sul nominale degli effetti

- GIORNI BANCA: 6 (5 gg. successivi alla scad. + giorno di negoziazione)
- GIORNI BANCA PER EFFETTI A "VISTA": 16 gg. (10 gg. + 6 gg. banca)

BREVITA' per scad. brucianti:
- SU PIAZZA: 12 gg. (entro i 12 gg. dopo data negoz.)
- F. PIAZZA: 20 gg. (entro i 20 gg.)

VALUTE DI ACCREDITO

SCONTO EFFETTI: 1° giorno lavorativo successivo alla data di negoziazione
EFFETTI ACCREDITATI SBF (Salvo Buon Fine):

SCAD. A
- VISTA
  - SU PIAZZA: 15 gg. fissi dopo la data di negoziazione
  - F. PIAZZA: 25 gg. fissi dopo la data di negoziazione
- GIORNO FISSO
  - SU PIAZZA: 10 gg. fissi dopo la scadenza effetto
  - F. PIAZZA: 20 gg. fissi dopo la scadenza effetto

Si intende su PIAZZA se l'effetto è pagabile nello stesso Comune di residenza della Banca negoziatrice, con la sola eccezione di quelli domiciliati su altre banche della stessa piazza. Clausola legale di domiciliazione: PAGABILE PRESSO ...

per motivi di recupero indipendentemente dalla destinazione che si vuol dare al proprio portafoglio effetti.
In parole povere, è una vera operazione di banca, completa di ogni particolare tecnica-applicativa e completa applicazione del corrente cartello bancario.
Il programma, dunque, potrà essere utile a tutti coloro che hanno problemi di negoziazione effetti, sia per motivi di recupero che di controllo; gli elaboratori delle banche, come tutti, non sbagliano, ma gli addetti possono sbagliare, (per esperienza diretta posso dire che non capita raramente ...), si fa presto a digitare un mese errato o, addirittura, un anno errato.

Modalità operative

Il programma è solo BASIC, privo di richiami in linguaggio macchina, universale per i PET/CBM e VIC e facilmente adattabile su qualsiasi elaboratore operante in BASIC.
Dopo le opzioni iniziali, scelta di stampa solo video o su stampante e avendo definito l'operazione da affrontare, *SCONTO* o *VALUTA MEDIA*, il programma chiede la data dell'operazione, dato importante perchè corrisponde alla data di inizio conteggi e bisogna digitarla in un unico numero di 6 cifre, es: l'1 Gennaio 1983 lo si deve digitare *010183 ...... RETURN ...*
Il programma chiede poi la quantità di effetti da elaborare, e qui l'utente introdurrà il relativo numero.
Se trattasi di operazioni *SCONTO*, prima di passare agli input degli estremi degli effetti, il programma chiede ancora se trattasi di operazione di *RINNOVO* (battere *SI/NO*) quindi di nuovo il *TASSO* e basta battere ad esempio 25 per dire il 25%.

```
         10 - 1 - 83

   ACCREDITO SBF DI N. 4 EFFETTI

   N.   SCAD.         LIRE    P.ZA
    1   VISTA      1000000     F
    2   28  2 83    500000     P
    3   30  3 83    250000     P
    4   30  4 83    250000     F

   TOTALE ...L.  2000000

   SPESE E COMMISSIONI...L.  10500

   LA VALUTA MEDIA E':   6 / 3 / 83
         10 - 1 - 83

  SCONTO N. 4 EFF. AL TASSO DEL 25 %

  N.   SCAD.         LIRE      INT.   P.ZA
   1   VISTA      1000000     11112    F
   2   28  2 83    500000     19098    P
   3   30  3 83    250000     14757    P
   4   30  4 83    250000     20139    F

  TOT....L.  2000000 INT.L. 65106

  COMM. 10000 - PROVV. 5000 -SPESE 500

  TOTALE COMPETENZE L.  80606
```

Figura 2 - Due esempi di output: operazione sconto e accrediti SBF di una medesima operazione.

Effetti bancari

A questo punto viene visualizzato un modulo guida per l'introduzione degli estremi nei singoli effetti ed il ciclo viene ripetuto per N volte, pari al numero digitato per la quantità di effetti.
Il cursore resterà posizionato progressivamente sulla scadenza, importo e piazza di ogni effetto. Esaudire gli input come segue: battere *V* - se la scadenza è a *VISTA*; battere *GGMMAA* - sempre in unica cifra di 6 caratteri se trattasi di scadenza a giorno fisso; battere *P* - se l'effetto è su piazza; battere *F* - se l'effetto è fuori piazza.
Per la distinzione della piazza di pagamento si veda il quadro del cartello bancario. La piazza è importante per la negoziazione accredito SBF: comporta una differenza di ben 10 giorni di valuta, mentre per una operazione di sconto serve solo ai fini della verifica del diritto di brevità per gli effetti a scadenza bruciante.
Terminata l'introduzione degli estremi degli effetti, il programma procede autonomamente per la parte rimanente: verifica il diritto di brevità, computa giorni, commissioni, interessi, valuta media e visualizza, su video o stampante, il risultato finale.
Il programma ha una storia lunga (*e un tantino tormentata*, NdR). E' nato suddiviso in singoli programmi, uno per lo sconto e uno per la valuta media, entrambi in duplice versione: *VIDEO* e *PRINTER*. Successivamente sono stati fusi *SCONTO* e *VAL. MEDIA* ma sempre in duplice versione *VIDEO* e *PRINTER* per favorire eventuali adattamenti. Infine ho redatto l'attuale versione con stampa optional, certamente più completa e anche per espresso desiderio della Redazione di **Bit**.

REMarks

180/530	*Parte del programma comune alle due sezioni operative;*
180/330	*Opzioni iniziali di stampa o non, e scelta di operazione, sconto o val. media; inizializzazioni e dimensionamento variabili A(24) per lettura data mesi e B$ (25) per gestione curs. verticale.*
340	*Input data di negoziazione; digitare GGMMAA in un numero intero di 6 cifre, es. 1 Genn. 1983 va battuto 010183;*
350	*Separazione della stringa data Da$ in tre sottocampi di giorno G, mese E e anno A1;*
360/460	*Doppie routine di stampa, VIDEO o PRINTER, secondo la scelta fatta inizialmente; tutte le routine di stampa saranno ripetute in duplice per le due versioni di stampa;*
400	*Input N = quantità di effetti da elaborare, routine comune ai due settori;*
470/530	*Visualizzazione modulo guida per i vari input degli estremi dei singoli effetti; anche questa resta comune ai due settori di programma;*
550/1100	*SEZIONE SCONTO EFFETTI.*
580	*Primo salto in subroutine di riga 1740/1810 per il computo dei giorni dalla data di riferimento 0101 anno in corso, alla data di negoziazione;*
590/950	*Loop di elaborazione effetti che verrà ripetuto per N effetti e che ha due chiusure, una in riga 890 per l'output printer, e l'altro in riga 950 per l'output solo su video;*
600/660	*Input scadenza eff. Z $ e distinzione di scad. a VISTA o scad. giorno fisso e computo giorni con aggiunta giorni del cartello bancario;*
670	*Input importo di un singolo effetto;*
680	*Input della piazza di pagamento: P$ = ad F se fuori piazza e = a P se su piazza;*
700/710	*Computo interesse di un singolo effetto;*
720/770	*Verifica diritto di brevità, computo commissioni, verifica minimo di interesse (L. 2000);*
790/880	*Output di ogni singola operazione, formattata per stampanti Commodore: 3022/4022;*
910/940	*Stesso output su video;*
990	*Computo della commissione di rinnovo: 1/4, cioè lo 0,25% sul nominale degli effetti;*
1000/1100	*Le due versioni di output per il risultato finale dell'operazione;*
1140/1720	*SEZIONE VALUTA MEDIA per accredito effetti SBF*
1140/1580	*È quasi simile alla sezione sconto, differisce per un minor input, diversa formattazione di stampa e diverso computo;*
1590/1630	*La parte tecnicamente più interessante del programma, trasforma il numero giorni ottenuto in data GGMMAA rappresentante la VALUTA MEDIA dell'operazione; la routine può essere fatta in modi più lunghi e complessi: la scelta fatta si è ispirata alla semplicità e possiede buona affidabilità.*
1740/1810	*Subroutine per computo giorni, chiamata in causa alla data di negoziazione ed in tutte le date di scadenza effetti, escluse quelle a VISTA; i DATA sono i giorni di ogni singolo mese ed è ripetuta per permettere il calcolo particolare di giorni per date che si presentano a cavallo di fine anno. Per gli anni bisestili, volendo sempre calcoli esatti, occorre ritoccare da 28 a 29 il DATA relativo a Febbraio: sempre per non appesantire il programma ho evitato istruzioni per la verifica di anno bisestile.*

```
 10 REM**********************************
 20 REM*      NEGOZIAZIONE EFFETTI      *
 30 REM*                                *
 40 REM*            BIT 1983            *
 50 REM**********************************
180 CLR:C$="████████████████████████████
":DIMB$(25)
190 PRINT"██             OPZIONE DI STAMP
A"
200 PRINT"█   █1█= SU VIDEO          █2█
= SU PRINTER"
210 GETX$:IFX$=""THEN210
220 IFX$<>"1"ANDX$<>"2"THEN210
230 IFX$="1"THEN280
240 PRINT"█ BENE, PREPARARE LA STAMPAN
TE !!";:FORJ=1TO1000:NEXT
250 OPEN1,4
260 OPEN2,4,1
270 OPEN3,4,2
280 DIMA(24):FORI=1TO25:B$(I)=LEFT$(C$
,I):NEXT
290 PRINT"█";TAB(5)"NEGOZIAZIONE EFFET
TI IN BANCA"
300 PRINT:PRINT"              QUALE OPERAZ
IONE  ?"
310 PRINTB$(10)" SCONTO EFF. = █1█
    EFF. SBF = █2█"
320 GETY$:IFY$=""THEN320
330 IFY$<>"1"ANDY$<>"2"THEN 320
340 PRINT"█":INPUT"DATA DI NEGOZIAZION
E GGMMAA█████████";DA$
350 G=VAL(LEFT$(DA$,2)):E=VAL(MID$(DA$
,3,2)):A1=VAL(RIGHT$(DA$,2))
360 PRINT"█          DATA DI NEGOZIAZI
ONE "
370 IFX$="1"THEN390
380 PRINT#1:PRINT#1,TAB(10)G;"-"E"-";A
1
390 PRINT:PRINTTAB(14)G;"-"E"-";A1
400 PRINTB$(5);:INPUT" N.EFF.";N:
410 IFY$="2"THENGOTO480
420 PRINTB$(5);TAB(11);:INPUT"    RINN.
";R$
430 PRINTB$(5)TAB(26);:INPUT"TASSO ";T
T
440 IFX$="1"THEN470
450 PRINT#1:PRINT#1,"SCONTO N."N"EFF.
AL TASSO DEL"TT"%"
460 PRINT#1:PRINT#1,"N.";SPC(2)"SCAD."
;SPC(6)"LIRE";SPC(7)"INT.  P.ZA"
470 REM **** VISUAL.MODULO GUIDA ****
480 FORC=1TO39:PRINT"▔";:NEXTC
490 PRINTB$(7)"   SCAD.EFF
    PIAZZA "
500 PRINTB$(8)" VISTA O G,M.     LIRE
P=SP F=FP"
510 FORC=1TO39:PRINT".";:NEXTC
520 PRINTB$(6);:FORC=1TO6
530 PRINT"|"TAB(13)"|"TAB(26)"|"TAB(38
)" |":NEXTC
```

```
 540 IFY$="2"THENGOTO1140
 550 REM **************************
 560 REM * SEZIONE SCONTO EFFETTI *
 570 REM **************************
 580 M=E:GOSUB1760:G=S+G
 590 FORK=1 TO N
 600 PRINTB$(12)SPC(2);:INPUT Z$
 610 IFZ$="V"THENZ$="VISTA":GOTO660
 620 IFZ$<>"V"THENX=VAL(LEFT$(Z$,2)):Y=
VAL(MID$(Z$,3,2)):A2=VAL(RIGHT$(Z$,2))
 630 M=Y:GOSUB1750:H=X+S
 640 Q=H-G:IFE>YOR(E=YANDG>=H)THENQ=(36
5-G)+H
 650 Q=Q+6:GOTO670
 660 Q=16
 670 PRINTB$(12)TAB(14);:INPUT L
 680 PRINTB$(12)TAB(31);:INPUT P$
 690 PRINTB$(12);:FORI=1TO120:PRINT" ";
:NEXT
 700 U=L*Q:D=36000/TT:Z=INT(U/D) :REM C
ALCOLO INTERESSE
 710 Z1=U/D:IFZ1>ZTHENZ=Z+1
 720 REM VERIFICA DIRITTO DI BREVITA'
 730 IFP$="P"ANDQ<18THENB=3000
 740 IFP$="F"ANDQ<26THENB=3000
 750 IFZ$="VISTA"THENB=0
 760 IFZ<2000THENZ=2000
 770 T=T+L:W=W+Z:B1=B1+B:B=0
 780 PRINT:PRINT:PRINT
 790 FORC=1TO39:PRINTCHR$(20);:NEXT
 800 IFX$="1"THEN910
 810 IFZ$<>"VISTA"THEN860
 820 G$="99  AAAAA      99999999  9999999
   A"
 830 PRINT#3,G$
 840 PRINT#2,K,Z$,CHR$(29),L,Z,P$
 850 GOTO890
 860 F$="99  99 99 99 99999999  9999999
   A"
 870 PRINT#3,F$
 880 PRINT#2,K,X,Y,A2,L,Z,P$
 890 NEXTK
 900 GOTO960
 910 IFZ$<>"VISTA"THEN930
 920 PRINTB$(14)K;Z$;:GOTO940
 930 PRINTB$(14)K;X;Y;A2;
 940 PRINTTAB(14)L;SPC(1)P$;SPC(2)"INT.
";Z
 950 NEXTK
 960 FORJ=1TO2000:NEXT
 970 PRINTB$(14);:FORI=1TO38:PRINT" ";:
NEXT
 980 EE=500:CM=2500*N+B1
 990 IFR$="SI"THEN O=INT(T*.0025):REM P
ROVV.RINNOVO EFF.
 1000 IFX$="1"THEN1060
 1010 PRINT#1:PRINT#1,"TOT.....L.  ";T;
 1020 PRINT#1,"INT.L."W
 1030 PRINT#1:PRINT#1,"COMM."CM"- PROVV
.";O;"-SPESE";EE
 1040 R=W+EE+CM+O
 1050 PRINT#1:PRINT#1,"TOTALE COMPETENZ
E L.";R:PRINT#1:PRINT#1
 1060 PRINT:PRINT"  TOT.L.";T;"   *TOT
.INT.."W
 1070 PRINT"▯COMM."CM"- PROVV."O"-  SPE
SE"EE
 1080 R=W+EE+CM+O
 1090 PRINT"▯TOTALE COMPETENZE L.";R
 1100 CLOSE1:CLOSE2:GOTO1700
 1110 REM ************************
 1120 REM * SEZIONE VALUTA MEDIA *
 1130 REM ************************
 1140 M=E:GOSUB1760:G=S+G-1
 1150 IFX$="1"THEN1190
 1160 PRINT#1
 1170 PRINT#1,"ACCREDITO ▯SBF■ DI N.";N
;"EFFETTI":PRINT#1
 1180 PRINT#1,"N.  SCAD.         LIRE
 P.ZA"
 1190 FORK=1 TO N
 1200 PRINTB$(12)SPC(2);:INPUT Z$
 1210 IFZ$="V"THENZ$="VISTA":GOTO1260
 1220 IFZ$<>"V"THENX=VAL(LEFT$(Z$,2)):Y
=VAL(MID$(Z$,3,2)):A2=VAL(RIGHT$(Z$,2))
 1230 M=Y:GOSUB1750:H=S+X
 1240 Q=H-G:IFE>YOR(Y=EANDG>=H)THENQ=(3
65-G)+H
 1250 Q=Q+20:GOTO1270
 1260 Q=25
 1270 PRINTB$(12)TAB(14);:INPUT L
 1280 PRINTB$(12)TAB(31);:INPUT P$
 1290 PRINTB$(12);:FORI=1TO120:PRINT" "
;:NEXT
 1300 IFP$="P"THENQ=Q-10
 1310 U=L*Q :REM ** CALCOLO NUMERO..**
 1320 T=T+L:W=W+U
 1330 IFP$="P"ANDQ<22THENB=3000
 1340 IFP$="F"ANDQ<40THENB=3000
 1350 IFZ$="VISTA"THENB=0
 1360 B1=B1+B:B=0
 1370 IFX$="1"THEN1480
 1380 IFZ$<>"VISTA"THEN1430
 1390 F$="99  AAAAA        99999999    AA
"
 1400 PRINT#3,F$
 1410 PRINT#2,K,Z$,CHR$(29),L,P$
 1420 GOTO1460
 1430 G$="99  99 99 99     99999999    AA
"
 1440 PRINT#3,G$
 1450 PRINT#2,K,X,Y,A2,L,P$
 1460 NEXTK
 1470 GOTO1540
 1480 IFZ$<>"VISTA"THEN1500
 1490 PRINTB$(14)K;Z$;:GOTO1510
 1500 PRINTB$(14)K;X;Y;A2;
 1510 PRINTSPC(2)L,P$
 1520 NEXTK
 1530 FORJ=1TO1500:NEXT
 1540 CM=2500*N+500+B1
 1550 PRINTB$(14);:FORI=1TO38:PRINT" "
:NEXT
 1560 Z=INT(W/T) :REM ** GIORNI DI VALU
TA MEDIA **
 1570 PRINT:PRINT"TOT.LIRE=";T
 1580 REM **CONVERS.GIORNI IN DATA VALU
TA MEDIA**
 1590 G=G+Z:R=1+INT(G/30):M=R:GOSUB1750
:G=G-S+1
 1600 IFG>31THENG=G-31:R=R+1:GOTO1620
 1610 IFG>30ANDR=4ORR=6ORR=9ORR=11THENG
=G-30:R=R+1:GOTO1620
 1615 IFG>28ANDR=2THENG=G-28:R=R+1
 1620 IFR>12ANDA2=A1THENR=R-12:A2=A2+1
 1630 IFR>12ANDA2>A1THENR=R-13
 1640 IFX$="1"THEN1680
 1650 PRINT#1:PRINT#1,"TOTALE ...L. "T
 1660 PRINT#1:PRINT#1,"SPESE E COMMISSI
ONI...L.";CM
 1670 PRINT#1:PRINT#1,"LA VALUTA MEDIA
E': ";G"/"R"/"A2:PRINT#1:PRINT#1
 1680 PRINT:PRINT"SPESE E COMMISSIONI..
.L.";CM
 1690 PRINT:PRINT"LA VALUTA MEDIA E':
";G"/"R"/"A2;:CLOSE1:CLOSE2
 1700 PRINTB$(23)"ALTRE OPERAZIONI ? ▯<
S/N>■"
 1710 GETY$:IFY$=""THEN1710
 1720 IFY$<>"S"THEN END
 1730 GOTO180
 1740 REM **SUBRSAVE.PER COMPUTO GIORNI
**
 1750 RESTORE
 1760 S=0
 1770 FORI=0TOM:READA(I):NEXTI
```

CBM

Effetti bancari

Seguito listato effetti bancari

```
 1780 FORJ=1TOM:S=S+A(J-1):NEXTJ
 1790 DATA0,31,28,31,30,31,30,31,31,30,
31,30,31
 1800 DATA0,31,28,31,30,31,30,31,31,30,
31,30,31
 1810 RETURN
```

Lista simboli grafici

```
  180 : 1   HOME =CHR$(19)
        25   CRSR↑ =CHR$(17)

  190 : 1   SHIFT HOME =CHR$(147)
        1   CRSR↑ =CHR$(17)

  200 : 1   CRSR↑ =CHR$(17)
        1   REVERSE =CHR$(18)
        1   SHIFT REVERSE =CHR$(146)
        1   REVERSE =CHR$(18)
        1   SHIFT REVERSE =CHR$(146)

  240 : 1   CRSR↑ =CHR$(17)

  290 : 1   SHIFT HOME =CHR$(147)

  310 : 1   REVERSE =CHR$(18)
        1   SHIFT REVERSE =CHR$(146)
        1   REVERSE =CHR$(18)
        1   SHIFT REVERSE =CHR$(146)

  340 : 1   SHIFT HOME =CHR$(147)
        8   SHIFT CRSR← =CHR$(157)

  360 : 1   SHIFT HOME =CHR$(147)

 1070 : 1   CRSR↑ =CHR$(17)

 1090 : 1   CRSR↑ =CHR$(17)

 1170 : 1   REVERSE =CHR$(18)
        1   SHIFT REVERSE =CHR$(146)

 1700 : 1   REVERSE =CHR$(18)
        1   SHIFT REVERSE =CHR$(146)
```

milano 7-11 giugno 1983

BIT USA 83: L'UNICA MOSTRA IN ITALIA DEDICATA AL PERSONAL COMPUTER

Il 1983 è l'anno del Personal Computer e il BIT USA 83 - organizzato dal Centro Commerciale Americano - è la più importante occasione per conoscere da vicino questo affascinante "protagonista" dei nostri giorni.
Il Personal Computer, infatti, non è più unicamente uno strumento di lavoro, ma va sempre più acquistando una sua precisa funzione in ogni settore, dallo studio, allo svago, all'organizzazione domestica, ecc.
Per questi motivi, il BIT USA 83 - unica manifestazione a riunire e presentare tutta la più qualificata produzione MADE IN USA del settore - è in grado di offrire spunti di indubbio interesse sia agli operatori che al pubblico.
Una 'Computer School' - organizzata con la collaborazione del Gruppo Editoriale Jackson - sarà, infatti, a disposizione dei visitatori per consentire, sotto la guida di esperti, un reale incontro con le mille e mille possibilità e capacità del 'Personal Computer', mentre un importante seminario verrà dedicato - nei giorni 8 e 9 giugno - al tema "Come aumentare la produttività in ogni reparto della vostra azienda".

Orario della mostra 9-18 continuato

CENTRO COMMERCIALE AMERICANO

Via Gattamelata 5,
20149 Milano Tel. (02) 46.96.451
Telex 330208 USIMC-I

Breakpoint!

Simulare una partita a tennis sull'esperienza di dati reali: un esempio per predire la possibile vittoria del giocatore preferito.

di U. G. Barzaghi

Questa volta il programma è ispirato non al titolo di un film, ma a quello di un romanzo: "Gioco!" (titolo originale, appunto, "Breakpoint!") di William Brinkley, incentrato sul debutto di un giovane tennista agli Internazionali degli Stati Uniti di Forest Hills, dove questa manifestazione si è disputata fino al suo trasferimento a Flushing Meadows.
Gli appassionati di tennis avranno avuto modo di notare come la traduzione in italiano del titolo del libro non sia stata molto accurata: breakpoint infatti, non indica la chiusura di un gioco (game), ma la possibile conquista del gioco da parte del giocatore che risponde al servizio, e che, quindi, dovrebbe, in teoria, trovarsi in stato di inferiorità; il quale crea, aggiudicandosi il punto, una frattura (break), in classifica, tra se e l'avversario. Si può quindi affermare che il breakpoint rappresenta la fase cruciale di una partita.

Tennis & computer

Il sottotitolo di questo programma è A.T.P., che, come i tifosi di tennis sapranno, è la sigla dell'associazione professionistica di tennis, responsabile di quell'assurdità sia tecnica che sportiva che va sotto il nome di "Classifica del computer". Nel mio programma, invece, A.T.P. sta per la sigla inglese A Tennis Predictor; vale a dire "un programma di previsione a carattere tennistico".
Torniamo, comunque, per un attimo sul discorso dell'impiego del computer nello stilare le classifiche di merito dell'associazione tennis professionistico. Il tennis è stato, probabilmente, il primo sport ad utilizzare il computer per qualcosa di diverso dal controllare e riordinare i tempi in gara di più concorrenti (impiego rapidamente diffusosi nelle gare di sci e nelle corse automobilistiche e motociclistiche; nelle quali l'uso del calcolatore è, in fondo, "logico", oltre che limitato all'utilizzo di una funzione di "Sort", allo scopo di riordinare la classica).
Nel caso del tennis, il problema che si desiderava risolvere era rappresentato dall'attribuzione di un ipotetico titolo di "Miglior tennista dell'anno". Prima dell'introduzione della classifica del computer questo titolo veniva conteso dai vincitori dei principali tornei sui vari tipi di fondo: Wimbledon (erba), Internazionali di Francia (terra battuta), U.S. Open (Deco Turf-Asphalt, una superficie artificiale) ed i giocatori che avevano offerto le migliori prestazioni in Coppa Davis, ritenuta un test molto probante, un tempo, quando tutti i maggiori giocatori vi partecipavano, poiché viene disputata su superfici ed in situazioni, sia climatiche che... di tifo, molto differenti.
La situazione era però destinata a fossilizzarsi in polemiche sterili. Le due principali riviste di tennis, "World Tennis" e "Tennis", compilano classifiche annuali, indipendenti l'una dall'altra, in base ai giudizi dei rispettivi esperti. L'A.T.P., per eliminare il fattore legato al giudizio di un esperto, o presunto tale, più o meno prevenuto nei confronti di questo o quel tennista, introdusse il computer e pubblicò, e pubblica tuttora, una classifica quindicinale basata sui risultati complessivi di ciascun giocatore negli ultimi dodici mesi. Trascorsi dodici mesi da una brutta sconfitta, capita quindi che un tennista risalga d'incanto anche trenta posizioni non appena il computer "dimentica" tale infausto episodio. I tennisti, come afferma Bjorn Borg nel suo libro "La mia vita il mio tennis" (Bjorn Borg, My life and game as told to Gene Scott) - Sperling & Kupfer editori, Milano, sono diventati ormai molto abili nel battere il computer e spesso riescono a calcolare come migliorare decisamente la loro posizione in classifica semplicemente sospendendo l'attività per una settimana. L'ostilità del campione svedese nei confronti della classifica stilata con il computer può forse essere spiegata con il fatto che, mentre egli è stato classificato primo al mondo per quattro anni secondo la votazione di fine d'anno dei giocatori, solo nel 1979 il computer è stato d'accordo con i suoi colleghi.
L'antipatia per il computer e la sua classifica non e però limitata al solo Borg, tra i tennisti professionisti, ed è piuttosto diffusa anche tra i semplici appassionati; quel che ne è derivato è un fenomeno piuttosto bizzarro e, decisamente, irrazionale, benché, purtroppo, piuttosto diffuso: la colpa viene unanimemente attribuita al computer, quasi fosse lui a decidere autonomamente; mentre, ovviamente, il problema è costituito da un'errata scelta della soluzione.
La questione, innanzi tutto, riguarda quali gare debbano essere prese in considerazione per stilare la classifica suddetta. Trattandosi di tennis professionistico, e venendo redatta negli Stati Uniti, si è, inevitabilmente, finito col privilegiare i tornei che fanno parte del circuito del W.C.T. - World Championship Tennis, che si disputano, in larga maggioranza, appunto negli Stati Uniti. Era quindi prevedibile che, da una simile classifica, ne uscissero avvantaggiati i giocatori che partecipano al maggior numero di queste manifestazioni, guarda caso per lo più americani, a scapito di giocatori che si riservano, come Borg (che, per chi non l'avesse ancora capito, è il mio favorito), di disputare un numero ridotto di tornei di eccellenza, preferendo la qualità del gioco alla quantità. Per una somma di tutte queste ragioni, la classifica del computer ha assegnato la palma di miglior giocatore dell'anno, negli anni scorsi, a tennisti che privilegiavano un'attività esasperata: in ordine di tempo, Jimmy Connors, John Paul Mc Enroe ed Ivan Lendl.
Anche l'associazione tennistica statunitense ha una sua classifica mensile calcolata con un computer, la quale, però, viene praticamente ignorata da tutti; questa classifica, comunque, viene utilizzata per stabilire la resa dei giocatori statunitensi alla fine di ogni anno.

Le caratteristiche del programma

Unici dati su cui si basa l'intera simulazione, sono le vittorie ottenute dai due giocatori in precedenti incontri con lo stesso avversario. Ho scelto questo dato come indicativo per diverse ragioni: innanzi tutto è molto semplice da reperire; la stampa sportiva, anche quella quotidiana, non solo quella specializzata, riporta, in genere, prima degli incontri più importanti, l'andamento dei precedenti scontri tra due tennisti.

In secondo luogo è relativamente facile da interpretare, anche ai fini implementativi; è inutile appesantire la simulazione, tenendo conto di fattori specifici, come la percentuale di prime e seconde palle servite da un giocatore o l'efficacia percentuale di questo o quel colpo nei confronti di un determinato avversario, se, poi, l'effetto di questi fattori sul risultato della simulazione è minimo e non aggiunge niente in realismo e precisione.

Più utile sarebbe, invece, ma è un genere di dato che difficilmente viene riportato - utilizzare invece del numero di incontri vinti da ciascuno dei due giocatori in precedenti edizioni dello stesso confronto, il numero di set che ognuno dei due si è aggiudicato negli incontri suddetti o, polverizzando ulteriormente l'analisi, il numero di game o, addirittura, di punti.

Questo concetto di "tennis percentuale" è espresso con grande precisione nel libro di Borg da me citato, e l'autore vi dedica addirittura un paragrafo all'interno di un capitolo, significativamente intitolato "Mandare la

REMarks

6	*Apertura del minuscolo (POKE59468,14), dimensionamento del vettore di gestione video e della matrice contenente i dati riguardanti lo svolgimento della gara. La matrice è dimensionata per eccesso sul numero massimo di set (5), ma viene utilizzata limitatamente ai set effettivamente disputati. Per ogni set:* *Il campo 1) contiene il punteggio del set per il primo concorrente; il campo 2) il punteggio del secondo concorrente; il campo 3) il punteggio dell'eventuale tie-break per il primo concorrente, il 4) per il secondo; i campi 5) e 6) contengono, rispettivamente, i minuti ed i secondi di durata del set.*
7	*Inizializzazione di alcune variabili:* *H0, M0 - secondi e minuti iniziali.* *FG - semaforo dell'avvenuto accesso alla subroutine 6000.* *E, per entrambi i concorrenti:* *S() - Punteggio nel set in corso.* *G() - Punteggio del gioco in corso.* *M() - Andamento del match (set vinti).* *PS() - Numero di palle complessivamente servite.* *DF() - Totale doppi falli.* *AC() - Numero di "ace" totalizzati.* *BP() - Break a favore.* *P() - punti totalizzati.* *SS() - set-point a disposizione.* *PP() - prime palle correttamente servite.* *SP() - seconde palle correttamente servite.* *MP() - match-point necessari per chiudere l'incontro.*
10	*Creazione del vettore di gestione cursore video.*
15-30	*Stampa dell'intestazione.*
40-90	*Acquisizione dati: i nomi del primo e del secondo giocatore (minuscolo ammesso) sono significativi fino al quindicesimo carattere. Il numero di vittorie è riferito a precedenti incontri CON LO STESSO AVVERSARIO. In particolare:*
75-84	*Scelta della formula più opportuna per il tie-break. La 1, consente di simulare incontri di singolare di Coppa Davis; la 2 singolari di tornei classici internazionali (Wimbledon, Roland Garros, Flushing Meadows); la 3 incontri di esibizione e partite del circuito W.C.T., finali dei Masters compresi.*
85-87	*Numero di set su cui si disputa l'incontro: 3 o 5, la risposta richiesta è rappresentata da una*

delle due cifre suddette.

90 — *Vengono ricavati, dai dati introdotti, gli indici necessari per lo svolgimento dell'incontro: ad una base di probabilità del 50% viene aggiunta un'opportuna frazione della differenza tra le partite vinte e le partite perse.*

100-185 — *Creazione del tabellone e del campo di gioco. Per la parte grafica si è utilizzata la consueta formula POKE32767+NR★40+NC, CD dove NR è il numero di riga, NC il numero di colonna, rispettivamente contenuti nei limiti 1-24 e 1-40, e CD è il numero di POKE corrispondente al carattere che si desidera visualizzare.*

100 — *Con la tecnica suddetta, viene creato il fondo del tabellone con il carattere "spazio" in negativo (numero di POKE 160).*

110 — *Vengono opportunamente inseriti sul tabellone i nomi dei giocatori e creati gli spazi che conterranno i risultati dei set precedenti.*

120 — *Vengono azzerati, sul tabellone, sia il punteggio del set che del game in corso.*

130 — *Vengono azzerati gli orologi.*

140-185 — *Viene creato il campo.*

190 — *Viene generato casualmente il giocatore al servizio.*

195 — *Viene inizializzato il contatore del numero di game disputati.*

200-290 — *Queste righe sono responsabili dell'accesso alle varie subroutine che permettono la regolare disputa dell'incontro. Rappresentano la centrale di controllo dell'intera simulazione. Questa tecnica, costituita da più sottoprogrammi richiamati in un ordine opportuno da una sezione centrale, è molto diffusa e fa parte, come chi ha già avuto a che fare con qualcuno dei miei programmi ben sa, del mio stile di programmazione. Una tecnica del genere, sapientemente usata, permette di liberarsi subito dei conflitti di competenza e dei problemi di terminazione o dei bivii che si vengono a formare nel flusso della lavorazione, demandando alle subroutine i problemi specifici: attenti però a non cadere in una eccessiva disgregazione dei problemi (subroutine è, si, bello; ma prima o poi determinate cose vanno comunque fatte).*

200 — *Viene fatto partire il cronometro. Vengono inizializzate alcune variabili contenenti i nomi dei giocatori e delle stringhe opportune, che servono a cancellare da video la vecchia posizione del nome di un giocatore, dopo una battuta (la stringa infatti contiene i caratteri che formano le righe di fondo del campo).*

210 — *Si accede alla subroutine che stabilisce in base al numero di set quanti set sono necessari per aggiudicarsi la partita. Quindi, se sono stati disputati tutti i set desiderati, si accede alla subroutine di fine match.*
N.B. In questa sezione del programma si possono notare numerosi accessi alla SUBroutine 2000, che rappresenta l'aggiornamento di entrambi gli orologi. Nel resto del testo non citerò più l'accesso a questa subroutine.

220 — *Si accede alla subroutine che controlla il termine del set.*

230 — *Se si è giunti alla conquista del gioco da parte di uno dei due giocatori si accede alla subroutine che analizza e segnala l'avvenimento.*

240 — *Si accede alle subroutine che segnalano sul tabellone il punteggio del game in corso e, se necessario, effettuano il cambio di campo. La inizializzazione a 2 della variabile P indica che il giocatore al servizio ha a disposizione due palle. La variabile GG (giocatore in possesso della palla) viene uguagliata a GS (giocatore al servizio).*

245 — *Si accede alle subroutine che analizzano se il punto da giocare è decisivo per la conquista del set (set-point) o, addirittura, del match (match-point). Si accede quindi alla subroutine che comunica a quale giocatore spetta il servizio (all'inizio di ogni set). Se il nome del giocatore al servizio è già stato comunicato si accede al sottoprogramma che segnala il punteggio parziale del gioco in corso.*

250 — *Viene aggiornato il totale delle palle servite dal giocatore al servizio. Quindi si trasferisce il controllo alla subroutine che si occupa di evidenziare il giocatore che ha a che fare con la palla. Quindi, in serie, si accede alla subroutine che genera numeri pseudo-casuali ed a quella che gestisce la prima palla di servizio.*
Se la variabile P è ancora uguale a 2 (il servizio ha toccato il nastro), si ritorna all'inizio della riga, poiché il giocatore ha a disposizione ancora due palle di servizio.

255 — *Se la variabile P è uguale a 9 (palla in gioco) si accede direttamente alle righe che si occupano degli scambi di gioco.*

257 — *Se P vale 10 ("ace") si accede alla subroutine che aggiorna il punteggio del gioco in corso. Quindi si ricicla, poiché il servizio è vincente.*

260 — *Analoga alla riga 250. In questo caso però si accede alla subroutine che si occupa della seconda palla di servizio. Se P è, in uscita, uguale ad 1 ("net"), si ricicla sulla stessa riga.*

265-266 — *Analoghe alle 255-257.*

267 — *Se P è uguale a 0 ("doppio fallo"), si aggiorna il punteggio e si ricicla.*

270-280 — *Rappresentano gli scambi di gioco e costituiscono il cuore della simulazione. In 270, dopo aver provveduto ad evidenziare il giocatore che risponde al servizio (SUBroutine 16000), si analizza se la risposta del giocatore stesso è al di sopra o al di sotto del suo margine di errore, inizializzato in 11000.*
Il numero casuale K, derivato dalla SUBroutine 3000, viene diminuito di 25 per prolungare gli scambi abbassando la percentuale di errore del giocatore. Aumentando in valore assoluto questo bonus gli scambi verranno ulteriormente prolungati; diminuendo o, addirittura, sopprimendolo, gli scambi saranno notevolmente abbreviati e la simulazione guadagnerà in rapidità, pur perdendo qualcosa in fatto di spettacolarità.
In 280 l'operazione viene ripetuta, se il giocatore che risponde al servizio è andato esente da errori, per il giocatore al servizio.

290 — *Se entrambi i giocatori hanno avuto successo nello scambio, si ricicla. Il ciclo viene interrotto dall'errore di uno dei due, il che porta all'aggiornamento del punteggio da parte della subroutine 15000.*

300-400 — *Tabellone riassuntivo delle statistiche relative all'incontro.*

300 — *Si accede alla subroutine di temporizzazione (1500) per permettere un'ultima occhiata al tabellone ed al campo da gioco.*

305 — *Si stampa l'intestazione, con i nomi dei due giocatori.*

310 — *Numero di ace totalizzati da ciascun giocatore nel corso dell'incontro.*

315 — *Si calcolano due variabili in modo da incolonnare opportunamente i dati immediatamente seguenti.*

320 — *Numero di doppi falli commessi dai concorrenti.*

330 — *Numero di break ottenuti.*

335 — *Viene ricavata la percentuale di prime palle di servizio sul totale delle palle servite dal primo giocatore.*

340 *La percentuale viene stampata.*

345-350 *Analoghe alle righe 335-340 per il secondo giocatore.*

355-370 *Analoghe alle righe 335-350 per le seconde palle di servizio.*

380 *Punti ottenuti dai due giocatori. Per punti si intendono quelli vinti all'interno dei singoli game.*

390 *Match-point che ciascuno dei giocatori ha avuto a disposizione. Per match-point si intende il punto che, se vinto, permette al giocatore che lo ha ottenuto di aggiudicarsi l'incontro.*

400 *Set-point. Il set-point è, invece, la palla che permette di chiudere il set in proprio favore.*

410-480 *Tabellone riassuntivo dell'andamento dell'incontro.*

410 *Dopo aver dato, tramite l'accesso alla routine di temporizzazione all'interno di un ciclo da 1 a 10, l'opportunità all'utente di leggere la prima parte della tabella, si stampa l'intestazione.*

415-430 *Andamento dei set. N.B. vengono mostrati solo i set effettivamente giocati.*

435 *Se non si è giocato alcun tie-break, si salta la successiva sezione di programma.*

437-460 *Punteggi degli eventuali tie-break giocati.*

461-468 *Vengono opportunamente trattate le stringhe della durata dei singoli set e dell'intero incontro.*

460-480 *Display dei tempi suddetti.*

999 *Fine del programma. Indica il termine della vasta struttura di controllo di cui sopra.*

1000 *Subroutine di temporizzazione, nell'ambito della partita, tramite l'accesso alla subroutine 2000, di aggiornamento degli orologi.*

1500 *Subroutine di temporizzazione semplice.*

2000-2040 *Subroutine di aggiornamento del tabellone. In 2000-2010 vengono ricavate e stampate le stringhe contenenti i minuti ed i secondi trascorsi dall'inizio del match. In 2020-2040 viene calcolato e visualizzato il tempo trascorso dalla fine del set precedente.*

3000 *Subroutine generatrice di numeri casuali tra 0 e 100, estremi esclusi. L'indice della funzione RND è costituito dalla somma della variabile TI (che viene però inizializzata per essere utilizzata come orologio), aumentata di un valore T0 che è stato inizializzato prima di azzerare il cronometro.*

4000 *Subroutine di clearing della riga di schermo su cui escono i messaggi.*

5000 *Subroutine che comunica la conclusione del match.*

6000-6040 *Subroutine che analizza l'eventuale conclusione del set.*

6000 *Inizializzazione delle variabili contenenti i punteggi di un eventuale tie-break. Analisi sulla necessità di accedere alla subroutine che si occupa della disputa dei tie-break.*

6003 *Si accede alla subroutine che calcola il tempo di durata del set appena concluso e si inizializzano opportunamente le variabili H0 e M0 che contengono secondi e minuti in cui si è concluso il set, in base al cronometro generale.*

8006-6007 *Vengono opportunamente trattate le stringhe contenenti i risultati del set da visualizzare sul tabellone.*

6008 *Vengono memorizzati nei campi loro destinati dalla matrice SET(), sia i punteggi del set per entrambi i giocatori, sia quelli degli eventuali tie-break (se il tie-break non è stato necessario, i campi opportuni sono posti a zero, permettendo di testare sulla necessità di mostrarli nella tabella di riga 400-480).*

6010-6015 *Vengono visualizzati i risultati del set nel settore di tabellone loro destinato, e vengono azzerati i punteggi del set successivo e del game.*

6020-6030 *Viene aggiornato il punteggio dell'incontro.*

6040 *Viene decrementato il contatore dei set giocati e riazzerate le variabili contenenti il punteggio del set in corso. Infine, viene posto uguale a 1, la variabile-semaforo FG che segnala l'avvenuto passaggio dalla subroutine in questione.*

7000-7050 *Subroutine che segnala l'avvenuta conquista del gioco da parte di uno dei due giocatori. In particolare:*

7000 *Viene aggiornata la variabile indicante il numero di giochi disputati. Viene comunicata, in posizione opportuna, la conquista del gioco.*

7002 *Viene eseguito un test sul punteggio dei due concorrenti, per stabilire se il gioco è stato conquistato dal giocatore al servizio o dal giocatore che risponde al servizio, nel qual caso viene segnalato il break.*

7003 *Viene incrementato il contatore dei break per il giocatore che lo ha ottenuto.*

7005-7007 *Viene invertito il servizio.*

7010-7020 *Viene incrementato il punteggio del set in corso per il giocatore che ha conquistato il set.*

7030-7040 *Vengono opportunamente allestite le stringhe che devono comparire sul tabellone e vengono visualizzate in posizione opportuna.*

7050 *Le variabili indicanti i punteggi del gioco in corso vengono azzerate, sia logicamente all'interno del programma, che materialmente sul tabellone di gara.*

8000-8020 *Subroutine di aggiornamento del tabellone.*

8000-8010 *Vengono convertite in stringhe da visualizzare sul tabellone le variabili indicanti il punteggio del gioco in corso, fino al 40 pari.*

8015-8018 *Vengono tradotti in stringhe opportune i valori di punteggio parziale del gioco di 95 (che è stato utilizzato per indicare la situazione di vantaggio da parte di un giocatore) e di 99 (che indica la conquista del game).*

8020 *I punteggi di cui sopra vengono visualizzati.*

9000-9030 *Subroutine che effettua il cambio di campo.*

9000 *Il test evita che il cambio di campo venga effettuato se è già avvenuto.*

9003 *Se il cambio di campo si rende necessario, le stringhe contenenti i nomi dei due giocatori vengono scambiate tra loro.*

9005 *Le stringhe vengono opportunamente accorciate (max. 9 caratteri) per poter trovare posto sul campo di gioco.*

9010-9030 *Le stringhe in questione vengono visualizzate in posizioni opportune a seconda dell'alternarsi della posizione di battuta, regolata dal test di riga 9010.*

10000-10030 *Subroutine di gestione della prima palla di servizio.*

10000 *Si accede alla subroutine che attribuisce le percentuali di efficacia dei colpi (o di incidenza degli errori, se preferite).*

10005 *Se il numero pseudo-casuale generato prima di accedere alla subroutine è superiore alla percentule di efficacia del giocatore al servizio aumentata di un bonus (+15), viene segnalata la necessità per il giocatore di ricorrere alla seconda palla di servizio.*

10020 *Si accede alla subroutine che si occupa di generare casualmente i net e gli ace, dopo aver opportunamente inizializzato la variabile KK, che, all'interno della subroutine suddetta, serve da indice di probabilità per entrambi i test che riguardano net e ace. Con questo artificio, si ottengono un numero sia di net che di ace, inferiore in caso di seconda palla di servizio rispetto alla prima palla di servizio. Il test in uscita dalla subroutine, aggiorna il conteggio delle prime palle di servizio nel caso*

testa oltre la rete".
In questo paragrafo, "Il calcolo delle probabilità nella strategia del tennis", Borg difende la sua tattica preferita, stare a fondo campo ed aspettare che l'avversario sbagli, sostenendo che il suo gioco di attesa costringe l'avversario a correre rischi. In sostanza Borg sostiene che il calcolo delle probabilità gioca a favore di quei giocatori che, come lui, sono forti negli scambi da fondo campo. Se un avversario come Roscoe Tanner, dice Borg, piazza venticinque ace di servizio in un incontro, probabilmente vince, ma non può farlo molto spesso. Il gioco di Borg lo costringe a giocare un tennis a bassa percentuale di probabilità.
Questa strategia "probabilistica" nel tennis si è venuta evolvendo nel tempo, come ogni cosa in ogni campo, del resto; un tempo, ad esempio, il modo più sicuro per aggiudicarsi un punto importante era giocare sul rovescio di un avversario, poiché erano ben pochi i giocatori in possesso di un rovescio veramente efficace, oggi, invece, ci sono decine di giocatori, tra i quali, ovviamente, tutti i migliori, che non aspettano altro che l'avversario spari sul loro rovescio quando è in gioco un punto importante.
Per Borg, invece, la questione non è tanto giocare sul diritto o sul rovescio; la sua definizione di "tennis percentuale" è "pazienza". Il suo unico scopo è di giocare una palla buona in più rispetto al suo avversario. Borg vuole indurre il suo avversario in errore o nell'esecuzione o nella scelta tattica del tipo di gioco da effettuare.
Nel tennis ad alto livello, dice Borg, non esistono punti di debolezza e di forza, ma solo forze comparate. Proprio di questo concetto di forze comparate mi sono appunto servito per realizzare il programma. Gli scambi di gioco sono rappresentati da dei test che verificano se un numero casuale rientra o no all'interno degli indici di efficacia percentuale dei due giocatori ricavati dai dati forniti da tastiera. In pratica lo scambio prosegue finché uno dei due giocatori non colpisce la sua "pallina" casuale (marca RND) al di sotto della percentuale indicata. Ciò permette di utlizzare il programma anche in quei casi in cui non esistono dati su cui basarsi o in cui i dati sono inconsistenti (nessun incontro giocato, o solo un numero limitato e con risultati poco plausibili). In questi casi, invece del numero di vittorie dei due giocatori, si possono inserire due valori (per esempio percentuali) delle probabilità di ciascun giocatore di aggiudicarsi l'incontro (ci si può basare sulle previsioni delle riviste specializzate, o semplicemente sul proprio giudizio).
Ciò permette di correggere anche quelle situazioni in cui tra due giocatori vi sono dati limitati ad uno o due incontri e con risultati avversi al naturale favorito: un paio d'anni fa capitò, ad esempio, che Borg, reduce da una serie di tornei sfibrante, perdesse in uno dei primi turni del torneo, se non erro, di Stoccarda contro uno sconosciuto tennista tedesco con cui non aveva mai giocato, e che occupava un posto oltre i primi cento della classifica del computer. Simulando il risultato di un successivo incontro tra gli stessi due giocatori, verrebbe quindi concesso un vantaggio, seppur lieve, a quello che è, in realtà, il più debole dei due. Inconveniente, questo, destinato, comunque, a non presentarsi molto spesso, poiché, in genere, i migliori giocatori si incontrano spesso tra loro e, di conseguenza, il materiale a disposizione, relativo agli incontri precedenti, è molto abbondante.
Per eliminare altri inconvenienti provocati da problemi di reperimento dei dati ho preferito rinunciare ad un'attribuzione proporzionale degli indici di merito per i due giocatori (vale a dire,

	in cui la variabile P sia uguale a 9 (palla in gioco) o uguale a 10 (ace).
11000-11010	*Subroutine che attribuisce alle variabili-indice di probabilità di errore, il valore legato, come si è detto, al numero di vittorie ottenute da ciascun giocatore. Al giocatore al servizio viene concesso un handicap dato dalla differenza - in valore assoluto - tra le partite vinte e le partite perse aumentata di cinque. Ciò per diminuire il numero di break.*
12000-12030	*Analoga alla 10000-10030, per la seconda palla di servizio. Da notare:*
12005-12010	*Il bonus maggiorato (+35), per aver un numero di doppi falli limitato (nessun giocatore rischia inutilmente di commettere doppio fallo). In caso di doppio fallo, l'aggiornamento della variabile relativa (DF) per il giocatore al servizio (GS).*
12020	*L'inizializzazione ad un valore minore della variabile KK, rispetto a quanto fatto per la subroutine "prima palla di servizio". L'aggiornamento della variabile che tiene conto delle seconde palle servite correttamente, se il servizio è andato a buon segno.*
13000-13020	*Subroutine che determina quanti set occorre vincere per aggiudicarsi l'incontro, sul totale dei set su cui si gioca la partita.*
14000-14010	*Questa subroutine genera casualmente i net e gli ace sia sulla prima che sulla seconda palla di servizio. La differenza è data dal valore con cui viene inizializzata la variabile KK, su cui sono basati i test di riga 14000-14005 prima di accedere alla subroutine in questione: ciò permette di aver più net ed ace sulla prima palla di servizio, generalmente più forzata, rispetto alla seconda palla. Da notare, in riga 14003, la variabile P che viene posta a 10, per segnalare l'ace e, in 14004, l'aggiornamento della variabile che conta, appunto, i servizi vincenti. In 14005, infine, la variabile P viene posta uguale a 9, per indicare che la palla è semplicemente in gioco, nel caso in cui non si tratti né di ace, né di net, né di fallo.*
15000-15080	*Questa subroutine analizza ed attribuisce il punteggio, in base all'andamento dell'incontro ed al punteggio precedente. Ciò si è reso necessario sia per il particolare tipo di punteggio utilizzato dal tennis (che, sia detto per inciso, sembra derivi dagli handicap, in piedi, che, nell'antico gioco della palla corda, venivano concessi da chi vinceva il punto all'avversario), sia per permettere l'analisi di casi come quelli in cui si vengono a trovare due giocatori giunti ai vantaggi per l'attribuzione del gioco.*
15000	*Viene aggiornata la variabile NG (punti giocati) su cui ci si basa per stabilire la posizione di battuta.*
15020	*Viene aggiornato il conteggio dei punti ottenuti.*
15025-15040	*Il punteggio del gioco viene aggiornato nei casi, diciamo così, normali.*
15050	*Se entrambi i concorrenti erano sul 40, viene segnalato il vantaggio (=95) del giocatore che ha conquistato il punto.*
15060	*Se l'avversario del giocatore che ha conquistato il punto si trovava in vantaggio la situazione viene riportata sul 40 pari.*
15070	*Se il giocatore che ha conquistato il punto si trova sul 40 e in situazione di vantaggio rispetto al suo avversario (40-0 o 40-15 o 40-30), viene segnalata la conquista del gioco, ponendo la variabile a 99.*
15080	*Se il giocatore che ha conquistato il punto si trovava già in vantaggio (indicato dalla variabile posta a 95) rispetto all'avversario, si vede attribuire il gioco.*
16000-16047	*Questa subroutine provvede a visualizzare opportunamente, sia come disposizione sul rettangolo di gioco, in relazione alla posizione di battuta, sia come modalità di rappresentazione (normale o inversa, per sottolineare il possesso della palla), i nomi dei due giocatori.*
17000-17010	*Subroutine che realizza l'alternanza degli scambi di gioco.*
18000-18010	*Subroutine che segnala il giocatore al servizio,*

	sia esplicitamente, sia attraverso il puntino posto accanto al nome del giocatore, sul tabellone.
19000-19050	*Questa subroutine segnala la situazione creatasi nel corso del game, dando la precedenza al punteggio del giocatore al servizio. Il programma distingue i seguenti casi:*
19010	*15, 30 o 40 pari. Così ho tradotto la forma inglese 15,30, 40 all.*
19015	*40 pari; indicata come parità (deuce), poiché precede i vantaggi.*
19020-19030	*Il vantaggio, rispettivamente, per il giocatore al servizio e per il suo avversario.*
20000-20010	*Questa subroutine stabilisce, a partire dal numero (1 o 2) che contraddistingue il giocatore al servizio, il numero del suo avversario (GA); per poterlo utilizzare in vari punti del programma.*
21000-21150	*Subroutine di gestione del tie-break.*
21000	*Il flusso del programma viene indirizzato alla riga opportuna in base all'esame del valore della variabile TB, che rappresenta la scelta dell'utente riguardo alle modalità del tie - break.*
	La struttura usata (ON TB GOTO ...,...), permette l'accesso alle righe seguenti a seconda del valore indicato: se si è scelta la prima modalità (chiamiamola tipo "Coppa Davis"), vale a dire nessun utilizzo della formula del tie-break, l'accesso alla subroutine viene chiuso dal RETURN.
	Nel secondo caso (tie-break su ciascun set, tranne l'ultimo), viene eseguito un test tendente appunto a stabilire se il set interessato sia o no l'ultimo: in caso negativo, il programma procede con la subroutine di controllo della situazione di tie-break.
	Nel terzo ed ultimo caso (tie-break sul sei pari in ogni set), si accede direttamente al nucleo del sottoprogramma.
21030	*Viene segnalata la situazione ed inizializzate le variabili che fanno da semaforo all'accesso a questa sezione di programma.*
21035-21090	*Questa sezione di programma ricalca a grandi linee quella delle righe 200-290, sia come struttura sintattica che come funzione logica, anche se, naturalmente, adattata alle caratteristiche ed alle funzioni del tie-break. Altri punti di contatto, che questi due gruppi di righe presentano, sono rappresentati dal frequente accesso alla subroutine 2000, per tenere aggiornati i due orologi sul tabellone, ed anche dall'utilizzo di alcune subroutine, dai compiti specifici, ma di utilizzo comune ad entrambe le sezioni.*
21035	*Test di terminazione del tie-break e, in caso di risposta positiva, accesso alla sezione opportuna della subroutine.*
21040	*Si accede alle subroutine che segnalano il punteggio del tie-break sul tabellone e che, se è necessario, effettuano il cambio di campo che, sul tie-break, è ogni sei punti.*
21045	*Si accede alla subroutine che segnala gli eventuali set-point e match-point, quindi si segnala il giocatore al servizio o, in alternativa (GOSUB23000), la situazione del tie-break.*
21050-21090	*Questa sezione di programma ricalca esattamente le righe 250-290. L'unica differenza è rappresentata dall'accesso, per l'attribuzione del punto conquistato, alla subroutine 24000, che si occupa di analizzare il punteggio del tie-break, a causa del suo particolare andamento; invece che alla subroutine 15000, che si occupa dello svolgimento normale del gioco.*
21100-21120	*Viene analizzato il risultato del tie-break ed aggiornato di conseguenza il risultato del set.*
21130-21150	*Si accede alla stessa subroutine che regola la conquista dei giochi, (il tie-break, infatti, può essere considerato alla stessa stregua di un normale gioco, per lo meno agli effetti del punteggio), saltando la sezione della subroutine che analizza il caso di "break", poiché non avrebbe, in questo caso, senso.*
22000-22010	*Subroutine che si occupa di stampare in posi-*

dare il 25% delle probabilità ad un giocatore che ha vinto 2 incontri su 8 con un determinato avversario, ed il 75% all'avversario stesso) per una divisione delle probabilità che tenga conto anche del numero di incontri disputati: ad una base di percentuale del cinquanta per cento viene sommata una frazione della differenza tra il numero di partite vinte ed il numero di partite perse da ciascun giocatore. Ciò permette di eliminare, in gran parte, i difetti insiti in precedenti poco numerosi o insolitamente sbilanciati.

Per capire appieno la differenza tra i due sistemi, basta notare che con il metodo proporzionale ottengono il 100% delle probabilità di vittoria sia un giocatore che abbia vinto venti incontri su venti con un determinato avversario, sia un giocatore che abbia vinto, magari fortunosamente, l'unico incontro disputato; mentre con il metodo da me adottato i pronostici sono molto più sbilanciati nel primo caso che nel secondo; contribuendo, tra l'altro, a prolungare, spettacolarmente, gli scambi.

A questo stesso scopo ho provveduto ad inserire un indice opportuno nella routine degli scambi.

Per tornare all'aspetto pratico del programma, gli unici altri dati necessari, a parte i nomi dei due giocatori, sono il numero di set su cui viene disputato l'incontro (al meglio dei tre o dei cinque set) e il meccanismo dei tie-break.

Il tie-break è stato introdotto dal tennis professionistico per evitare che le partite si prolunghino eccessivamente: poiché, infatti, per aggiudicarsi il set è necessario conquistare sei giochi con un vantaggio di almeno due sull'avversario, sul sei pari, nei casi cioé in cui non c'erano stati break - o ve n'era stato un numero equilibrato - si proseguiva finché una situazione di break non si veniva a creare. Ciò comportava partite lunghissime con comprensibili effetti poco consoni con le esigenze del tennis-spettacolo professionistico, con rigidi orari e programmi da rispettare. Venne quindi introdotto il tie-break, che prevede che, sul sei pari, si giochino dodici punti, scambiando il servizio ogni due punti, fino a che uno dei due giocatori non si sia aggiudicato almeno sette punti, con un vantaggio di almeno due sull'avversario, in caso contrario si continua a giocare, finché la situazione di break non si viene a creare. Il tie-break, introdotto inizialmente dai tornei professionistici del W.C.T. è stato in seguito adottato dalla maggior parte dei tornei: a tutt'oggi la sola Coppa Davis non lo contempla tra le sue regole; tornei più importanti lo applicano con la sola eccezione dell'ultimo set quello decisivo, se si rende necessario disputarlo; Wimbledon, roccaforte del conservatorismo tennistico inglese, ha introdotto una sua personale versione del tie-break, sull'otto pari; ma ha recentemente capitolato, fatto salvo l'ultimo set.

Il programma permette di scegliere tra ciascun tipo di combinazione, in modo da poter simulare qualsiasi tipo di incontro: Coppa Davis, Wimbledon, tornei del W.C.T. ed esibizioni.

La simulazione mostra sul video una rappresentazione sintetica dell'andamento dell'incontro, sotto forma di un tabellone, sul tipo di quello di Wimbledon, che reca, a sinistra, il risultato dei set precedenti; al centro i nomi dei due concorrenti (max. 15 caratteri), seguiti dal punteggio parziale del set in corso e dal punteggio del game o del tie-break che si sta disputando. All'estrema destra del tabellone vengono mostrati, sovrapposti, due orologi: quello superiore indica il tempo trascorso dall'inizio dell'incontro, l'inferiore, il tempo di gioco del set in corso. Un puntino indica inoltre il giocatore cui spetta il servizio. Per quanto riguarda il gioco, nel caso in cui si arrivi ai vantaggi, que-

```
BREAKPOINT

 1 REM ********************************
 2 REM *       BREAKPOINT (ATP)       *
 3 REM *                              *
 4 REM *           BIT 1983           *
 5 REM ********************************
 6 POKE59468,14:DIMW$(25),SET(5,6):PRIN
T"▯"
 7 H0=0:M0=0:FG=0:FORK=1TO2:S(K)=0:G(K)
=0:M(K)=0:PS(K)=0
 8 DF(K)=0:AC(K)=0:BP(K)=0:P(K)=0:SS(K)
=0:PP(K)=0:SP(K)=0:MP(K)=0:NEXT
 10 W$="▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮":FOR
W=1TO25:W$(W)=LEFT$(W$,W):NEXT
 15 PRINTW$(3)TAB(14)"|REAKPOINT!"
 20 PRINTW$(5)TAB(17)"▮▮▮ ▮▮▮ ▮▮▮":PRIN
TW$(10)TAB(10)"▮▮▮"
 30 PRINTW$(12)TAB(10)"▮|▮ENNIS":PRINTW
$(14)TAB(10)"▮▮▮REDICTOR":GOSUB1500
 37 REM *********************
 38 REM * ACQUISIZIONE DATI *
 39 REM *********************
 40 PRINT"▯":INPUT"/OME DEL PRIMO GIOCA
TORE";G$:G$(1)=LEFT$(G$+"              ",15
)
 50 INPUT"/OME DEL SECONDO GIOCATORE";G
$:G$(2)=LEFT$(G$+"              ",15)
 60 PRINT"▮/UMERO VITTORIE PRIMO GIOCAT
ORE";:INPUTV(1)
 70 PRINT"/UMERO VITTORIE SECONDO GIOCA
TORE";:INPUTV(2)
 75 PRINT"▮▮| IE-BREAK -"
 80 PRINT"▮1 - IN NESSUN SET":PRINT"2 -
 IN TUTTI I SETS, TRANNE L'ULTIMO"
 83 PRINT"3 - IN TUTTI I SETS":PRINT"▮"
;:INPUTTB
 84 IFTB<1ORTB>3THENPRINT"▯▯▯▯":GOTO83
 85 PRINT"▮▮▮▮L MEGLIO DEI TRE SETS (3)
 O CINQUE SETS (5)";:INPUTNS:SN=NS
 87 IFNS<>5ANDNS<>3THENGOTO85
 90 I(1)=50+(V(1)-V(2))/3:I(2)=50+(V(2)
-V(1))/3:PRINT"▯"
 97 REM ********************************
 98 REM * TABELLONE E CAMPO DA GIOCO *
 99 REM ********************************
 100 FORI=1TO200:POKE32767+I,160:NEXT:P
RINTW$(2)TAB(11)G$(1)
 110 PRINTW$(4)TAB(11)G$(2):FORI=1TO2:F
ORJ=1TO5:POKE32807+J*2+(I-1)*80,32:NEXT
 120 NEXT:PRINTW$(2)TAB(27)"0"TAB(29)"
0":PRINTW$(4)TAB(27)"0"TAB(29)" 0"
 130 PRINTW$(2)TAB(34)"00"TAB(37)"00":P
RINTW$(4)TAB(34)"00"TAB(37)"00"
 140 FORJ=0TO80STEP80:FORI=1TO31:POKE33
011+I+J,98:POKE33531+I+J,226:NEXT:NEXT
 150 FORI=0TO13:POKE33051+I*40,225:POKE
33083+I*40,97:NEXT:POKE33011,108
 160 POKE33043,123:POKE33643,126:POKE33
611,124:FORI=1TO17:POKE33298+I,111
 170 POKE33338+I,119:NEXT:FORI=0TO15:PO
KE33027+I*40,102:NEXT
 180 FORI=0TO9:POKE33138+I*40,225:POKE3
3156+I*40,97:NEXT
 185 POKE32987,98:POKE33667,226
 190 T0=TI:GOSUB3000:GS$=G$(1):GS=1:IFK
>49.99THENGS$=G$(2):GS=2
 195 NG=1
 197 REM *********
 198 REM * MATCH *
 199 REM *********
 200 TI$="000000":Q1$="     ▮▮      ":Q2$
="   ▮      ":N1$=G$(1):N2$=G$(2):GF=0
 210 GOSUB13000
 220 GOSUB2000:GOSUB6000:IFFG=1THENFG=0
:GOTO210
 230 GOSUB2000:IFG(1)=99ORG(2)=99THENGO
SUB7000:GOTO220
 240 GOSUB2000:GOSUB8000:GOSUB9000:P=2:
GG=GS
 245 GOSUB25000:GOSUB18000:IFYU=1THENGO
SUB19000
 250 SS(GS)=SS(GS)+1:GOSUB2000:GOSUB160
00:GOSUB3000:GOSUB10000:IFP=2THEN250
 255 GOSUB2000:IFP=9THEN270
 257 GOSUB2000:IFP=10THENGOSUB17000:GOS
UB15000:GOTO230
 260 SS(GS)=SS(GS)+1:GOSUB2000:GOSUB160
00:GOSUB3000:GOSUB12000:IFP=1THEN260
 265 GOSUB2000:IFP=9THEN270
 266 GOSUB2000:IFP=10THENGOSUB17000:GOS
UB15000:GOTO230
 267 GOSUB2000:IFP=0THENGOSUB15000:GOTO
230
 270 GOSUB17000:GOSUB16000:GOSUB3000:GO
SUB11000:IFK-25>W2THENGOSUB15000:GOTO23
0
 280 GOSUB17000:GOSUB16000:GOSUB3000:GO
SUB11000:IFK-25>W1THENGOSUB15000:GOTO23
0
 290 GOSUB2000:GOTO270
 297 REM *********
 298 REM * SCORE *
 299 REM *********
 300 GOSUB1500:PRINT"▯":PRINTW$(2)TAB(1
7)"▮#CORE▮"
 305 PRINTW$(5)TAB(13)"▮"LEFT$(G$(1),12
)"▮"TAB(28)"▮"LEFT$(G$(2),12)"▮"
 310 PRINTW$(7)"♠CES"TAB(19-LEN(STR$(AC
(1))))AC(1)TAB(34-LEN(STR$(AC(2))))AC(2
)
 315 C1=3-LEN(STR$(DF(1))):C2=3-LEN(STR
$(DF(2)))
 320 PRINTW$(9)"-OPPI FALLI"TAB(16+C2)D
F(1)TAB(31+C2)DF(2)
 330 PRINTW$(11)"|REAKS"TAB(17)BP(1)TAB
(32)BP(2):PRINTW$(12)"(A FAVORE)"
 335 PR=INT(PP(1)/SS(1)*10000)/100
 340 PRINTW$(14)"% PRIME PALLE"TAB(16)P
R"%";
 345 PR=INT(PP(2)/SS(2)*10000)/100
 350 PRINTTAB(31)PR"%":PRINTW$(15)" DI
SERVIZIO"
 355 PR=INT(SP(1)/SS(1)*10000)/100
 360 PRINTW$(17)"% SECONDE PALLE"TAB(16
)PR"%";
 365 PR=INT(SP(2)/SS(2)*10000)/100
 370 PRINTTAB(31)PR"%":PRINTW$(18)" DI
SERVIZIO"
 380 PRINTW$(20)"¬UNTI"TAB(16)P(1)TAB(3
1)P(2)
 390 PRINTW$(22)"\ATCH-POINTS"TAB(17)MP
(1)TAB(32)MP(2)
 400 PRINTW$(24)"#ET-POINTS"TAB(17)PS(1
)-MP(1)TAB(32)PS(2)-MP(2)
 410 FORY=1TO10:GOSUB1500:NEXT:PRINT"▯"
TAB(6)"|ABELLONE RIASSUNTIVO"
 415 FORI=1TOSN-NS:PRINTW$(4)"#ETS"TAB(
13+I*4)I"▮▮ ":NEXT
 420 PRINTW$(6)"▮"LEFT$(G$(1),12)"▮":PR
INTW$(8)"▮"LEFT$(G$(2),12)"▮"
 430 FORK=1TO2:FORH=1TOSN-NS:PRINTW$(4+
K*2)TAB(13+H*4)SET(H,K):NEXT:NEXT
 435 BT=0:FORK=1TO2:FORH=1TOSN-NS:BT=BT
+SET(H,K+2):NEXT:NEXT:IFBT=0THEN461
 437 PRINTW$(11)"| IE-BREAK"
 440 FORH=1TOSN-NS:FORK=1TO2:IFSET(H,3)
=0ANDSET(H,4)=0THENGOTO460
 450 PRINTW$(8+K*2)TAB(13+H*4)SET(H,K+2
)
 460 NEXT:NEXT
 461 HF=0:MF=0:FORK=1TOSN-NS:HF=HF+SET(
K,6):MF=MF+SET(K,5):NEXT
 462 FM=INT(HF/60):HF=HF-FM*60:MF=MF+FM
:HH=INT(MF/60):MF=MF-HH*60
```

```
 463 MF$=STR$(MF):HF$=STR$(HF):HH$=STR$
(HH):MF$=RIGHT$(MF$,LEN(MF$)-1)
 466 MF$=RIGHT$("00"+MF$,2):HF$=RIGHT$(
HF$,LEN(HF$)-1):HF$=RIGHT$("00"+HF$,2)
 467 IFHH<>0THENST$=HH$+"H"+MF$+"'"+HF$
+CHR$(34):GOTO469
 468 ST$=MF$+"'"+HF$+CHR$(34)
 469 PRINTW$(17)" ¬URATA DELL'INCONTRO
"ST$
 470 FORK=1TOSN-NS:S5$=STR$(SET(K,5)):S
6$=STR$(SET(K,6))
 475 S5$=RIGHT$("  "+S5$,2):S6$=RIGHT$(
S6$,LEN(S6$)-1):S6$=RIGHT$("00"+S6$,2)
 480 ST$=S5$+"'"+S6$+CHR$(34):PRINTW$(1
8+K)K"▌▛ ♦ET COMPLETATO IN "ST$:NEXT
 999 END
 1000 FORJ=1TO20:GOSUB2000:NEXT:RETURN
 1500 FORJ=1TO3000:NEXT:RETURN
 1997 REM ************
 1998 REM * OROLOGIO *
 1999 REM ************
 2000 S$=RIGHT$(TI$,2):S=VAL(S$):M$=MID
$(TI$,3,2):M=VAL(M$):H=VAL(LEFT$(TI$,2)
)
 2010 PRINTW$(2)TAB(34)M$TAB(37)S$:T=H*
3600+M*60+S-M0*60-H0:M=INT(T/60):S=T-M*
60
 2020 S$=RIGHT$("00"+RIGHT$(STR$(S),LEN
(STR$(S))-1),2)
 2030 M$=RIGHT$("00"+RIGHT$(STR$(M),LEN
(STR$(M))-1),2)
 2040 PRINTW$(4)TAB(34)M$TAB(37)S$:RETU
RN
 2997 REM **************************
 2998 REM * NUMERI PSEUDO CASUALI *
 2999 REM **************************
 3000 K=RND(TI+T0)*100:RETURN
 3997 REM ************************
 3998 REM * CLEARING DI RIGA 24 *
 3999 REM ************************
 4000 FORI=33688TO33728:POKEI,32:NEXT:R
ETURN
 4997 REM *********
 4998 REM * MATCH *
 4999 REM *********
 5000 PRINTW$(24)TAB(17)"╲ATCH!":GOSUB1
500:GOSUB4000:GOTO300
 5997 REM *******
 5998 REM * SET *
 5999 REM *******
 6000 TB(1)=0:TB(2)=0:Q=0:IFS(1)=6ANDS(
2)=6THENQ=1:GOSUB21000:IFGH=1THEN6050
 6001 IFQ=1ANDGH=0THEN6003
 6002 IF(S(1)<6ANDS(2)<6)ORABS(S(1)-S(2
))<2THENRETURN
 6003 GOSUB27000:M0=VAL(MID$(TI$,3,2)):
H0=VAL(RIGHT$(TI$,2))
 6005 PRINTW$(24)TAB(18)"♦ET!":GOSUB100
0
 6006 S1$=RIGHT$(STR$(S(1)),LEN(STR$(S(
1)))-1)
 6007 S2$=RIGHT$(STR$(S(2)),LEN(STR$(S(
2)))-1)
 6008 FORK=1TO2:SET(SN-NS+1,K)=S(K):SET
(SN-NS+1,K+2)=TB(K):NEXT
 6010 GOSUB4000:PRINTW$(2)TAB(1+(SN-NS)
*2)S1$:PRINTW$(4)TAB(1+(SN-NS)*2)S2$
 6015 PRINTW$(2)TAB(27)"0"TAB(29)" 0":P
RINTW$(4)TAB(27)"0"TAB(29)" 0"
 6020 IFS(1)>S(2)THENM(1)=M(1)+1:GOTO60
40
 6030 M(2)=M(2)+1
 6040 NS=NS-1:S(1)=0:S(2)=0:FG=1
 6050 GH=0:RETURN
 6997 REM ********
 6998 REM * GAME *
 6999 REM ********
 7000 GN=GN+1:GF=0:PRINTW$(24)TAB(17)"|
AME!":GOSUB1000:GOSUB4000:QG=0:NG=1
 7002 IFG(GG)>G(GS)THENPRINTW$(24)TAB(1
7)"|REAK!":GOSUB1000:GOSUB4000
 7003 IFG(GG)>G(GS)THENBP(GG)=BP(GG)+1
 7005 IFGS=1THENGS=2:GOTO7010
 7007 GS=1
 7010 IFG(1)>G(2)THENS(1)=S(1)+1:GOTO70
30
 7020 S(2)=S(2)+1
 7030 S1$=RIGHT$(STR$(S(1)),LEN(STR$(S(
1)))-1)
 7035 S2$=RIGHT$(STR$(S(2)),LEN(STR$(S(
2)))-1)
 7040 PRINTW$(2)TAB(27)S1$:PRINTW$(4)TA
B(27)S2$
 7050 PRINTW$(2)TAB(29)" 0":PRINTW$(4)T
AB(29)" 0":G(1)=0:G(2)=0:RETURN
 7997 REM ****************************
 7998 REM * AGGIORNAMENTO TABELLONE *
 7999 REM ****************************
 8000 H1$=STR$(G(1)):IFG(1)<>0THENH1$=R
IGHT$(H1$,LEN(H1$)-1)
 8010 H2$=STR$(G(2)):IFG(2)<>0THENH2$=R
IGHT$(H2$,LEN(H2$)-1)
 8015 IFG(1)=95THENH1$=" V"
 8016 IFG(2)=95THENH2$=" V"
 8017 IFG(1)=99THENH1$=" |"
 8018 IFG(2)=99THENH2$=" |"
 8020 PRINTW$(2)TAB(29)H1$:PRINTW$(4)TA
B(29)H2$:RETURN
 8997 REM ****************
 8998 REM * CAMBIO CAMPO *
 8999 REM ****************
 9000 IFGF=1THEN9005
 9003 IFGN<>INT(GN/2)*2THENT1$=N2$:N2$=
N1$:N1$=T1$:GF=1
 9005 N1$=LEFT$(N1$,9):N2$=LEFT$(N2$,9)
 9010 IFNG=INT(NG/2)*2THENGOTO9030
 9020 PRINTW$(18)N1$TAB(30)Q2$:PRINTW$(
11)Q1$TAB(30)N2$:RETURN
 9030 PRINTW$(11)N1$TAB(30)Q2$:PRINTW$(
18)Q1$TAB(30)N2$:RETURN
 9997 REM ****************
 9998 REM * PRIMA PALLA *
 9999 REM ****************
 10000 GOSUB11000
 10005 IFK>W1+15THENPRINTW$(24)TAB(13)"
♦ECONDA PALLA":GOSUB1000:GOSUB4000:P=P-
1
 10010 IFK>W1+15THENRETURN
 10020 KK=10:GOSUB14000:IFP=9ORP=10THEN
PP(GS)=PP(GS)+1
 10030 RETURN
 10997 REM *************************
 10998 REM * PERCENTUALI & BONUS *
 10999 REM *************************
 11000 W1=I(1)+ABS(V(1)-V(2)):W2=I(2):I
FGS=2THENW1=I(2)+ABS(V(1)-V(2)):W2=I(1)
 11010 W1=W1+5:RETURN
 11997 REM *****************
 11998 REM * SECONDA PALLA *
 11999 REM *****************
 12000 GOSUB11000
 12005 IFK>W1+35THENPRINTW$(24)TAB(14)"
¬OPPIO FALLO":GOSUB1000:GOSUB4000:P=P-1
 12010 IFK>W1+35THENDF(GS)=DF(GS)+1:RET
URN
 12020 KK=5:GOSUB14000:IFP=9ORP=10THENS
P(GS)=SP(GS)+1
 12030 RETURN
 13000 NN=3:IFSN=3THENNN=2
 13010 IFM(1)=NNORM(2)=NNTHENGOSUB5000
 13020 RETURN
 13997 REM ****************
 13998 REM * NETS & ACES *
 13999 REM ****************
 14000 GOSUB3000:IFK<KK/2THENPRINTW$(24
)TAB(18)"/ET!":GOSUB1000:GOSUB4000:RETU
RN
```

```
14003 IFK<KKTHENPRINTW$(24)TAB(18)"♠CE
!":GOSUB1000:GOSUB4000:P=10
 14004 IFK<KKTHENAC(GS)=AC(GS)+1:RETURN
 14005 IFK>KKTHENP=9
 14010 RETURN
 14997 REM ****************************
 14998 REM * ATTRIBUZIONE PUNTEGGIO *
 14999 REM ****************************
 15000 NG=NG+1:IFGG=1THENGG=2:GA=1:GOTO
15020
 15010 GA=2:GG=1
 15020 P(GG)=P(GG)+1
 15025 IFG(GG)=0THENG(GG)=15:RETURN
 15030 IFG(GG)=15THENG(GG)=30:RETURN
 15040 IFG(GG)=30THENG(GG)=40:RETURN
 15050 IFG(GG)=40ANDG(GA)=40THENG(GG)=9
5:RETURN
 15060 IFG(GG)=40ANDG(GA)=95THENG(GA)=4
0:RETURN
 15070 IFG(GG)=40ANDG(GA)<40THENG(GG)=9
9:RETURN
 15080 IFG(GG)=95THENG(GG)=99:RETURN
 15997 REM ******************
 15998 REM * ROUTINE SCAMBI *
 15999 REM ******************
 16000 N3$="▒"+LEFT$(N1$,9)+"■":N4$="▒"
+LEFT$(N2$,9)+"■"
 16010 IFNG=INT(NG/2)*2THENGOTO16030
 16015 IFGG<>GSTHEN16025
 16017 PRINTW$(11)Q1$TAB(30)N2$
 16020 PRINTW$(18)N3$TAB(30)Q2$:RETURN
 16025 PRINTW$(18)N1$TAB(30)Q2$
 16027 PRINTW$(11)Q1$TAB(30)N4$:RETURN
 16030 IFGG<>GSTHEN16045
 16035 PRINTW$(18)Q1$TAB(30)N2$
 16040 PRINTW$(11)N3$TAB(30)Q2$:RETURN
 16045 PRINTW$(11)N1$TAB(30)Q2$
 16047 PRINTW$(18)Q1$TAB(30)N4$:RETURN
 17000 IFGG=1THENGG=2:RETURN
 17010 GG=1:RETURN
 17997 REM ************************
 17998 REM * ALTERNANZA SERVIZI *
 17999 REM ************************
 18000 IFQG=1THENYU=1:RETURN
 18005 GOSUB20000:PRINTW$(GS*2)TAB(32)"
.":PRINTW$(GA*2)TAB(32)" "
 18010 PRINTW$(24)" ♠L SERVIZIO: "G$(GS
):GOSUB1000:GOSUB4000:QG=1:RETURN
 18997 REM ********************
 18998 REM * GIUDICE ARBITRO *
 18999 REM ********************
 19000 YU=0:GOSUB20000
```

```
 19005 IFG(GS)<>G(GA)THEN19020
 19010 IFG(GS)<>40THENPRINTW$(24)TAB(16
)"¬ARI"G(GS):GOSUB1000:GOSUB4000:RETURN
 19015 IFG(GS)=40THENPRINTW$(24)TAB(16)
"¬ARITA'":GOSUB1000:GOSUB4000:RETURN
 19020 IFG(GS)=95THENPRINTW$(24)"×ANTAG
GIO "G$(GS):GOSUB1000:GOSUB4000:RETURN
 19030 IFG(GA)=95THENPRINTW$(24)"×ANTAG
GIO "G$(GA):GOSUB1000:GOSUB4000:RETURN
 19040 Q$=STR$(G(GS))+" -"+STR$(G(GA)):
PRINTW$(24)TAB((39-LEN(Q$))/2)Q$
 19050 GOSUB1000:GOSUB4000:RETURN
 20000 IFGS=1THENGA=2:RETURN
 20010 GA=1:RETURN
 20097 REM **************
 20098 REM * TIE-BREAK *
 20099 REM **************
 21000 GH=0:ONTBGOTO21010,21020,21030
 21010 RETURN
 21020 IFNS=1THENGH=1:RETURN
 21030 PRINTW$(24)TAB(15)"|IE-BREAK!":G
OSUB1000:GOSUB4000:BT=1:JK=0
 21035 GOSUB2000:IF(TB(1)>=7ORTB(2)>=7)
ANDABS(TB(1)-TB(2))>=2THEN21100
 21040 GOSUB2000:GOSUB22000:GOSUB28000:
P=2:GG=GS
```

```
 21045 GOSUB26000:GOSUB18000:IFYU=1THEN
GOSUB23000
 21050 SS(GS)=SS(GS)+1:GOSUB2000:GOSUB1
6000:GOSUB3000:GOSUB10000:IFP=2THEN2105
0
 21055 GOSUB2000:IFP=9THEN21070
 21057 GOSUB2000:IFP=10THENGOSUB17000:G
OSUB24000:GOTO21035
 21060 SS(GS)=SS(GS)+1:GOSUB2000:GOSUB1
6000:GOSUB3000:GOSUB12000:IFP=1THEN2106
0
 21065 GOSUB2000:IFP=9THEN21070
 21066 GOSUB2000:IFP=10THENGOSUB17000:G
OSUB24000:GOTO21035
 21067 GOSUB2000:IFP=0THENGOSUB24000:GO
TO21035
 21070 GOSUB17000:GOSUB16000:GOSUB3000:
GOSUB11000
 21075 IFK-25>W2THENGOSUB24000:GOTO2103
5
 21080 GOSUB17000:GOSUB16000:GOSUB3000:
GOSUB11000
 21085 IFK-25>W1THENGOSUB24000:GOTO2103
5
 21090 GOSUB2000:GOTO21070
 21100 IFTB(1)<TB(2)THEN21120
 21110 S(1)=S(1)+1:GOTO21130
 21120 S(2)=S(2)+1
 21130 GOSUB7030:IFGS=1THENGS=2
 21150 GOSUB7030:QG=0:GOSUB22000:RETURN
 21997 REM ************************
 21998 REM * PUNTEGGIO TIE-BREAK *
 21999 REM ************************
 22000 FORJ=1TO2:H$=STR$(TB(J)):IFTB(J)
>9THENH$=RIGHT$(STR$(TB(J)),2)
 22010 PRINTW$(J*2)TAB(29)H$:NEXT:RETUR
N
 23000 YU=0:GOSUB20000
 23005 IFTB(GS)<>TB(GA)THEN23020
 23010 PRINTW$(24)TAB(16)TB(GS)"PARI":G
OSUB1000:GOSUB4000:RETURN
 23020 Q$=STR$(TB(GS))+" -"+STR$(TB(GA)
):PRINTW$(24)TAB((39-LEN(Q$))/2)Q$
 23030 GOSUB1000:GOSUB4000:RETURN
 24000 NG=NG+1:BT=BT+1:IFGG=1THENGG=2:G
A=1:GOTO24015
 24010 GA=2:GG=1
 24015 IFGG<>GSTHENPRINTW$(24)TAB(17)"|
REAK!":GOSUB1000:GOSUB4000
 24017 IFGG<>GSTHENBP(GG)=BP(GG)+1
 24020 P(GG)=P(GG)+1:TB(GG)=TB(GG)+1:IF
BT<2THENRETURN
 24025 BT=0:QG=0
 24030 IFGS=1THENGS=2:RETURN
 24040 GS=1:RETURN
 24997 REM **************
 24998 REM * SET-POINTS *
 24999 REM **************
 25000 SG=0:IF(S(1)+S(2))<5THENRETURN
 25005 IF(S(1)<5ANDS(2)<5)THENRETURN
 25010 IFS(1)=S(2)THENRETURN
 25020 IF((G(1)=95ANDG(2)=40)OR(G(1)=40
ANDG(2)<40))ANDS(1)>S(2)THENSG=1
 25030 IF((G(2)=95ANDG(1)=40)OR(G(2)=40
ANDG(1)<40))ANDS(2)>S(1)THENSG=2
 25040 IFSG=0THENRETURN
 25045 IFSG=2THENPS(2)=PS(2)+1:GOTO2505
0
 25047 PS(1)=PS(1)+1
 25050 M$="♠ET-POINT!":NN=3:IFSN=3THENN
```

```
N=2
 25057 REM ****************
 25058 REM * MATCH-POINTS *
 25059 REM ****************
 25060 IF(SG=1ANDM(1)=NN-1ANDM(2)<=NN-1
)THENM$="\ATCH-POINT!":MP(1)=MP(1)+1
 25065 IF(SG=2ANDM(2)=NN-1ANDM(1)<=NN-1
)THENM$="\ATCH-POINT!":MP(2)=MP(2)+1
 25070 PRINTW$(24)TAB((40-LEN(M$))/2)M$
:GOSUB1000:GOSUB4000:RETURN
 25997 REM **************
 25998 REM * SET-POINTS *
 25999 REM **************
 26000 IF(TB(1)+TB(2))<5THENRETURN
 26005 IF(TB(1)<6ANDTB(2)<6)THENRETURN
 26010 IFTB(1)=TB(2)THENRETURN
 26015 SG=1:IFTB(2)>TB(1)THENSG=2
 26020 GOSUB25045:RETURN
 27000 MF=VAL(MID$(TI$,3,2)):HF=VAL(RIG
HT$(TI$,2)):HH=VAL(LEFT$(TI$,2))
 27005 SE=HH*3600+MF*60+HF-M0*60-H0
 27010 SET(SN-NS+1,5)=INT(SE/60):SET(SN
-NS+1,6)=SE-SET(SN-NS+1,5)*60:RETURN
 28000 IFTB(1)+TB(2)=INT((TB(1)+TB(2))/
6)*6THENT1$=N2$:N2$=N1$:N1$=T1$
 28005 GOSUB9005:RETURN
```

sta situazione viene evidenziata con una "v" minuscola, in corrispondenza del giocatore che si trova in vantaggio.
Sotto al tabellone, viene visualizzata una rappresentazione del campo da gioco (per ragioni tecniche la rete è un pò spessa...), sul quale vengono fatti evoluire i nomi dei due giocatori (abbreviati, per ragioni di spazio, a nove caratteri). Il programma provvede ad amministrare le posizioni di battuta in maniera opportuna, così come i cambi di campo. Gli scambi vengono realizzati evidenziando in "reverse" il giocatore in possesso della palla.
Il programma provvede anche a notificare esplicitamente la situazione dell'incontro, segnalando a chi spetta il servizio, prima di ciascun gioco, e l'andamento del punteggio, sia nei game che in caso di tie-break, segnalando correttamente, come il miglior giudice-arbitro di Wimbledon, prima il punteggio del giocatore al servizio.
Il programma segnala inoltre i break, sia su di un set che nel tie-break, e le palle decisive: set-point e match-point.
Un discorso a parte va fatto per il servizio. Dopo aver segnalato il giocatore al servizio, all'inizio di un gioco, o il punteggio aggiornato, nel caso di giochi in corso; il programma accede ad una subroutine che stabilisce, tramite i soliti numeri casuali, l'effetto della prima palla di servizio; se il servizio va in rete o non cade nel settore di campo previsto, viene segnalato che il giocatore ha a disposizione una seconda palla di servizio. Se la subroutine viene superata senza problemi si accede ad un altro sottoprogramma, che testa la possibilità che la palla servita sia un "net" o un "ace": nel primo caso - la pallina ha toccato il nastro - si accede nuovamente alla subroutine che serve la prima palla, dopo aver segnalato il "net"; nel secondo caso - servizio vincente - il punteggio viene aggiornato, e si salta la sezione di programma che rappresenta lo scambio di gioco.
Un discorso analogo può

```
Lista simboli grafici

    6 : 1   SHIFT HOME =CHR$(147)

   10 : 1   HOME =CHR$(19)
       25    CRSR↑ =CHR$(17)

   20 : 1   REVERSE =CHR$(18)
        1   SHIFT REVERSE =CHR$(146)
        1   REVERSE =CHR$(18)
        1   SHIFT REVERSE =CHR$(146)
        1   REVERSE =CHR$(18)
        1   SHIFT REVERSE =CHR$(146)
        1   REVERSE =CHR$(18)
        1   SHIFT REVERSE =CHR$(146)

   30 : 1   REVERSE =CHR$(18)
        1   SHIFT REVERSE =CHR$(146)
        1   REVERSE =CHR$(18)
        1   SHIFT REVERSE =CHR$(146)

   40 : 1   SHIFT HOME =CHR$(147)

   60 : 1   CRSR↑ =CHR$(17)

   75 : 2   CRSR↑ =CHR$(17)

   80 : 1   CRSR↑ =CHR$(17)

   83 : 1   CRSR↑ =CHR$(17)

   84 : 4   SHIFT CRSR↑ =CHR$(145)

   85 : 3   CRSR↑ =CHR$(17)

   90 : 1   SHIFT HOME =CHR$(147)

  200 : 1   REVERSE =CHR$(18)
        1   SHIFT REVERSE =CHR$(146)

  300 : 1   SHIFT HOME =CHR$(147)
        1   REVERSE =CHR$(18)
        1   SHIFT REVERSE =CHR$(146)

  305 : 1   REVERSE =CHR$(18)
        1   SHIFT REVERSE =CHR$(146)
        1   REVERSE =CHR$(18)
        1   SHIFT REVERSE =CHR$(146)

  410 : 1   SHIFT HOME =CHR$(147)

  415 : 1   SHIFT CRSR← =CHR$(157)

  420 : 1   REVERSE =CHR$(18)
        1   SHIFT REVERSE =CHR$(146)
        1   REVERSE =CHR$(18)
        1   SHIFT REVERSE =CHR$(146)

  480 : 1   SHIFT CRSR← =CHR$(157)

16000 : 1   REVERSE =CHR$(18)
        1   SHIFT REVERSE =CHR$(146)
        1   REVERSE =CHR$(18)
        1   SHIFT REVERSE =CHR$(146)
```

zione opportuna e dopo aver opportunamente convertito le variabili numeriche in stringhe alfanumeriche, il punteggio del tie-break sul tabellone.

23000-23030 *Subroutine analoga alla 19000. In questo caso viene segnalato il punteggio del tie-break. Da notare che gli unici due casi che si presentano sono rappresentati dalla situazione di parità, che viene segnalata come 1, 2, 3, ecc. pari; ed il caso normale in cui viene segnalato il punteggio del giocatore al servizio, seguito da quello del suo avversario.*

24000-24040 *Questa subroutine analizza lo svolgimento del tie-break e si occupa di segnalare i break, limitatamente al tie-break. Da notare:*

24000 *Vengono aggiornati sia il conto dei giochi che il conto dei punti di tie-break disputati, ciò allo scopo di permettere un regolare alternarsi delle posizioni di battuta.*

24017 *Nel caso che il giocatore al servizio se lo sia visto strappare dall'avversario, viene aggiornata la variabile relativa.*

24020 *Vengono aggiornate le variabili che contengono il numero di punti conquistati ed il punteggio del tie-break per il giocatore che si è aggiudicato il punto. Viene quindi eseguito un test per stabilire se è necessario che i due giocatori si scambino il servizio (nel tie-break ciò accade ogni due punti).*

25000-25050 *Questa sezione di programma (di difficile messa a punto!), segnala i set-point che si verifichino nel corso del normale svolgimento di un set (quando, cioé, non si arriva al tie-break).*

25000 *Se la somma dei punteggi dei due concorrenti, relativa al set in corso, è inferiore a cinque, l'analisi viene interrotta, poiché non si possono presentare casi di set-point (casi, cioé, in cui il giocatore che si trova in vantaggio, aggiudicandosi il punto e, quindi, anche il gioco, si aggiudicherebbe il set o, addirittura (match-point), l'incontro).*

25005 *Se entrambi i giocatori hanno punteggio del set inferiore a 5, l'analisi viene interrotta.*

25010 *Se la situazione, all'interno del set in corso, è di parità, è inutile proseguire l'analisi, poiché, per conquistare il set, sarebbe comunque necessario un break.*

25020-25030 *Viene analizzata la situazione del gioco in corso per stabilire se uno dei due giocatori (quale viene eventualmente segnalato tramite la variabile SG), è nelle condizioni di conquistare il gioco ed il set insieme al punto.*

25045 *Viene incrementata la variabile contenente il numero di set-point che sono stati necessari a ciascun giocatore per aggiudicarsi il set.*

25060-25070 *Questa sezione, opportunamente accodata alla precedente, analizza se il caso di set-point non sia addirittura un caso di match-point. È stata, anche graficamente, separata dalla subroutine precedente, poiché viene utilizzata anche per l'analisi del caso di set-point nel corso del tie-break.*

26000-26020 *Analizza gli eventuali set-point che si dovessero presentare nel corso del tie-break.*

26000 *Analoga alla 25000, ma analizzando il punteggio del tie-break.*

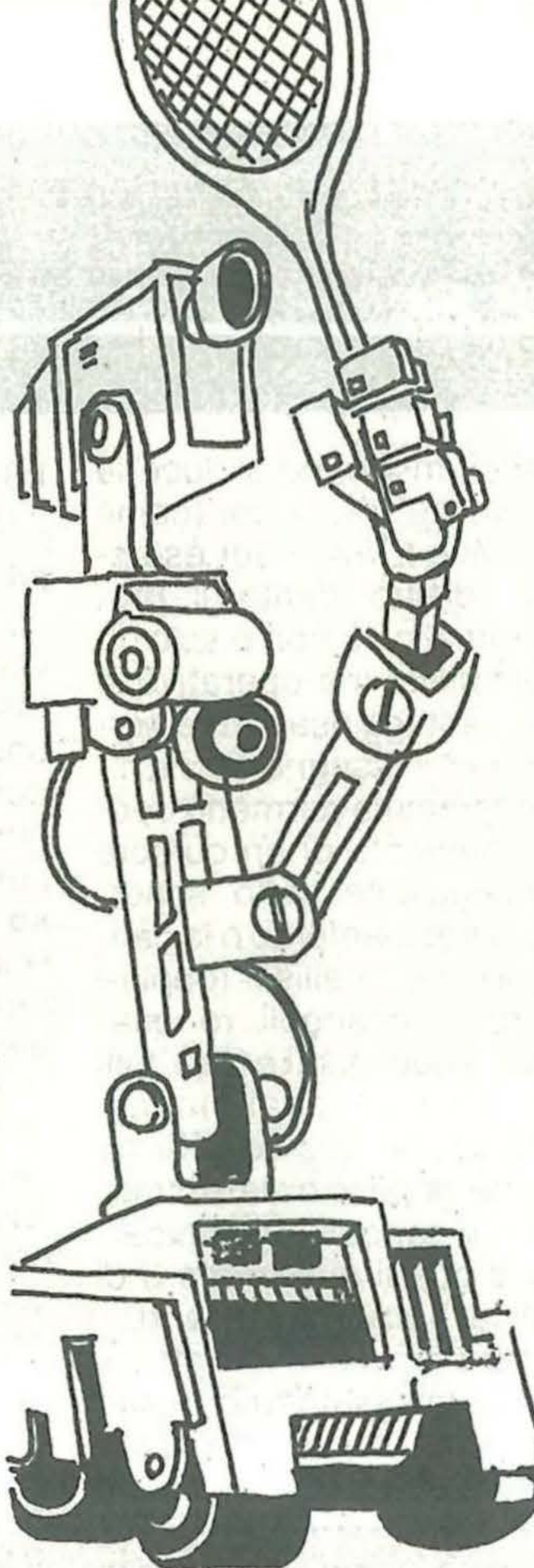
26005 *Analoga alla 25005.*

26010 *Provvede ad escludere il caso di parità.*

26015 *Stabilisce quale dei due giocatori abbia a disposizione una palla di set-point.*

26020 *Trasferisce la necessità di segnalare il set-point o l'eventuale match-point, da verificare, alla sezione di programma sopra descritta.*

27000-27010 *Provvede a scindere il tempo di durata del set conclusosi in minuti e secondi ed ad aggiornare le variabili relative.*

28000-28005 *Stabilisce se sia o no necessario il cambio di campo nel corso dei tie-break.*

essere fatto per la seconda palla di servizio, nel caso che si presenti questa evenienza, in questo caso però, se il servizio finisce in rete o fuori, viene segnalato il "doppio fallo", ed il punto viene attribuito al giocatore che risponde al servizio.

Tutti gli indici che regolano la incidenza percentuale di ciascuno degli effetti suddetti, sono stati inizializzati in modo da dare una percentuale di seconde palle, servizi vincenti, net e doppi falli plausibile.

Come detto, lo scambio continua finché uno dei due giocatori non colpisce la pallina al di sotto della sua percentuale di errore, perdendo il punto; gli scambi sono stati artificialmente prolungati, concedendo un bonus percentuale su di ogni palla, che può essere, se si vuole, modificato per accorciare od allungare l'incontro. Un vantaggio sulla propria percentuale di errore viene anche concesso al giocatore che detiene il servizio, come teoricamente dovrebbe essere, in modo da limitare i break. Al termine dell'incontro viene mostrato lo score dei due tennisti: vale a dire il numero di doppi falli commessi, di "ace" serviti, la percentuale di prime palle di servizio andata ad effetto sul totale delle palle servite, ed una analoga percentuale per le seconde palle (da notare che gli indici della subroutine che regola i "net" e gli "ace", vengono inizializzati in maniera differente a seconda che vi si acceda dopo la prima o dopo la seconda palla, in modo che, sia gli uni che gli altri siano piuttosto rari nel secondo caso: nessun buon tennista, infatti, a meno che non sia molto sicuro del proprio servizio o ridotto alla disperazione, forza mai la sua seconda palla di servizio). Vengono anche contati i punti realizzati ed i set-point e match-point che ciascun giocatore ha avuto a disposizione per chiudere un set o l'incontro. Quindi viene visualizzato un riassunto della partita, comprendente il punteggio di ciascun set, compreso il risultato di eventuali tie-break, con la durata dell'intera partita e di ogni singolo set.

I tempi sono, ovviamente, quelli effettivi della simulazione, e non corrispondono alla durata plausibile di un incontro.

Per chiudere vorrei citare due esempi di simulazione da me realizzati, per mettere alla prova la bontà del programma: nel primo caso ho inserito i dati riguardanti gli incontri tra Tanner e Borg prima della finale di Wimbledon del 1979; risultato: vittoria per Borg in cinque set con il punteggio di 6-7 (tie-break 2-7), 6-2, 5-7, 6-4, 6-2. Il punteggio effettivo fu: 6-7, 6-1, 3-6, 6-3, 6-4.

Un'anticipazione sul prossimo incontro fra Borg e Mc Enroe, al meglio dei cinque set (magari a Wimbledon, se non si incontrano prima); vittoria per Borg per 4-6, 7-5, 7-6 (tie-break 7-2), 6-3.

Desidero inoltre ringraziare il Dottor Sergio De Ceglie per la consulenza prestatami.

Disegno tecnico con HP 85

Uno strumento professionale per disegnare tramite semplici comandi da tastiera.

di V. Anselmo

La grafica è uno dei punti forti dell'HP-85, macchina apprezzata, oltre che per le capacità di elaborazione dei dati e per la sua compattezza, anche per il fatto di essere dotata di schermo, memoria di massa e stampante integrati.
L'esempio di programmazione che qui si propone è stato preparato per l'HP-85 con due scopi ben precisi, uno di studio e l'altro pratico: l'esame della lista delle istruzioni consentirà infatti al principiante di approfondire la meccanica dei comandi più espressamente grafici, mettendo in luce le differenze fra alcuni (come *DRAW* e *IDRAW* per esempio); d'altro canto il programma in sè non è solo dimostrativo, ma operativo, e può vantaggiosamente servire per il disegno tecnico.
Il programma permette il posizionamento di un cursore lampeggiante sullo schermo, il tracciamento o la cancellazione di ellissi (oppure cerchi), rettangoli (o quadrati), linee, punti e testi, tutti posizionabili esattamente ovunque si desideri. Si ha anche la possibilità di tracciare o cancellare semicerchi o quarti di cerchio o di ellisse, funzioni utili per disegnare archi in architettura, per lo smussamento di spigoli nel disegno meccanico e simili.
Una volta caricato il programma, come prima scelta propone quella per certe crocette di riferimento sullo schermo, dopodiche si passa al modo grafico. I tasti speciali (k1÷k8) hanno all'inizio le funzioni che seguono:
k1 (←) permette di spostare verso sinistra il cursore lampeggiante inizialmente posizionato al centro dell'area di tracciamento. Il cursore si sposta sempre con salti di 4 punti (pixel) alla volta, sia in orizzontale che in verticale, onde permettere un successivo più facile riposizionamento; per spostamenti successivi premere il cursore più volte (lo spostamento non si ripete automaticamente);
k2 (↑) sposta il cursore verso l'alto;
k3 (↓) sposta il cursore verso il basso;
k4 (→) sposta il cursore verso destra;
Osservazione: Qualora si fosse premuto per errore uno dei tasti di controllo del cursore dello schermo alfanumerico, premere CONT per continuare.
k5 (ELLISS) sceglie il modo "ellissi"; deve essere premuto una prima volta dopo aver posizionato il cursore in uno dei due punti in cui l'asse orizzontale dovrà intercettare la curva, e una seconda dopo aver posizionato il cursore in uno dei due punti in cui la curva sarà intercettata dall'asse verticale.
Come per i successivi comandi "rettangoli" e "linee", è il *secondo* comando quello che decide il tipo di figura. In altri termini si può premere indifferentemente uno qualsiasi dei tre comandi ("ellissi", "rettangoli" o "linee") per immettere il primo punto, che verrà segnato sullo schermo, e quindi, al momento di immettere il secondo punto, decidere sulla figura da tracciare. Per le ellissi intere (modo normale) è tuttavia importante ricordarsi di immettere prima il punto sull'asse orizzontale e poi quello sull'asse verticale, in modo che il programma possa capire di quale ellisse si tratta (v. figura 2). Se si è scelto con uno dei comandi accessibili tramite il tasto k8 *(COMANDI)* di tracciare un semicerchio o un quarto di cerchio fissandone il quadrante, si potranno immettere invece i due punti in un ordine qualsiasi, basta che uno sia su un asse (e sulla curva) e l'altro sull'altro asse (e sulla curva). L'ellisse o il cerchio sono tracciati con sedici segmenti corrispondenti alla corda dei successivi archi sottesi da un angolo di 22°,5 (360/16), che ruota in senso antiorario a partire da 0°. L'ellisse sarà sempre con gli assi paralleli ai bordi dell'area di tracciamento.
k6 (RETTAN) sceglie il modo "rettangoli". I due punti che si immettono dovranno trovarsi agli estremi della diagonale che unisce due vertici opposti del rettangolo (o quadrato). Come già detto a proposito dell'ellisse, è il secondo comando a selezionare la figura. Anche il quadrilatero avrà i lati paralleli ai bordi dell'area di tracciamento.
k7 (LINEE) Sceglie il modo "linee". Unisce due punti con un segmento di retta. Posizionando il cursore e premendo due volte "linee" si traccia un punto in quella posizione. Se la linea è l'ultima in basso e i due punti immessi sono posti agli estremi a sinistra e a destra, viene tracciata una linea che va oltre i bordi dell'area di tracciamento, per permettere di avere una base continua quando si voglia comporre un disegno architettonico che si sviluppi su più schermi in orizzontale (figura 1).
k8 (COMANDI) permette di passare ad un secondo gruppo di otto comandi. Essi costituiscono il sottogruppo seguente:

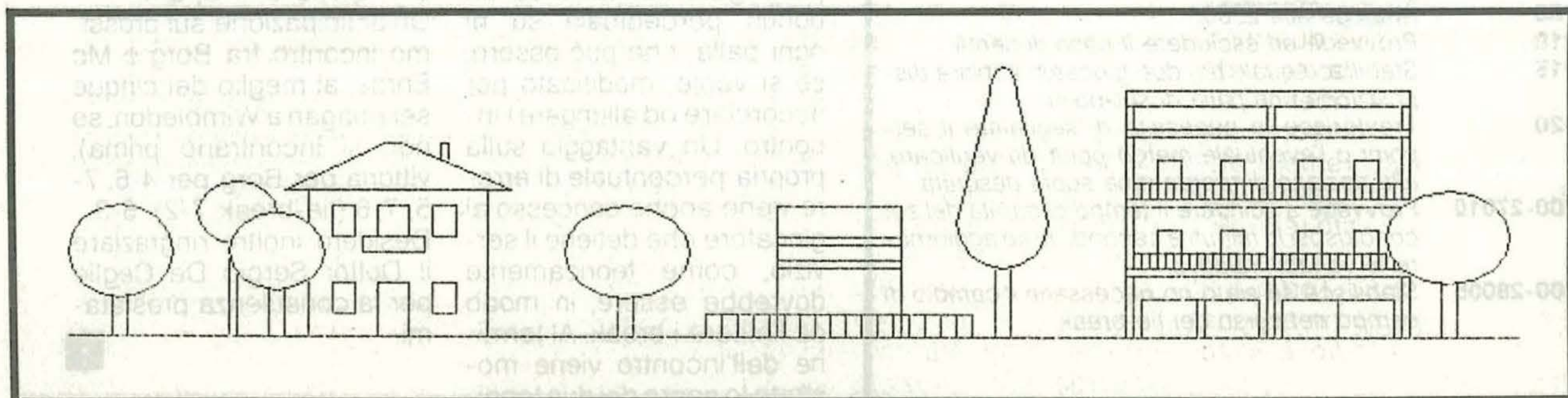



Figura 1 - Disegno tracciato su tre schermi successivi, che risultano uniti nella stampa.

k1 *(S A/S)* posiziona la concavità a sinistra predisponendo per il tracciamento di un semicerchio, oppure in alto a sinistra se si è in modo "quarti di cerchio". Torna al grafico, nel quale si dovranno immettere i due punti col comando "ellissi".
k2 *(A B/S)* posiziona la concavità in alto per i semicerchi, o in basso a sinistra per i quarti. Torna al grafico.
k3 *(B B/D)* posiziona la concavità in basso per i semicerchi, o in basso a destra per i quarti. Torna al grafico.
k4 *(D A/D)* posiziona la concavità a destra per i semicerchi, o in alto a destra per i quarti. Torna al grafico.
Nota:
dopo aver tracciato un semicerchio o un quarto di cerchio, il programma torna al modo normale (tracciamento dell'intera ellisse).
k5 *(TESTO)* permette di scrivere un testo sul grafico. La prima lettera sarà posizionata esattamente nella posizione del cursore (se si batte il segno +, il punto centrale del + corrisponderà al punto centrale del cursore lampeggiante). Torna al grafico.
k6 *(COPIA)* permette di copiare il grafico dopo avervi cancellato la parte inferiore relativa ai comandi. Torna al grafico cancellando tutti i dati tracciati in precedenza.
k7 *(QUARTI)* sceglie il modo "quarti di cerchio o di ellisse". Fa comparire la scritta *QUARTI* di conferma dell'opzione e resta in attesa della scelta del quadrante.
k8 *(CANCELLA)* predispone per la cancellazione e torna al grafico. Per cancellare una figura, impostare successivamente due punti, come quando la si è disegnata. Per cancellare un testo, portare il cursore sulla prima lettera, premere ancora *COMANDI*, poi *TESTO* e immettere il testo (che deve essere identico a quello da cancellare). Torna nel modo normale non appena la figura o il testo è stato cancellato.
Nota:
se una figura cancellata ne incrociava un'altra, risulteranno cancellati anche i punti in cui le due figure si sovrapponevano. In tal caso sarà necessario ripristinare l'integrità della figura danneggiata, ritracciandola.
Per annullare il comando *CANCELLA* se premuto per errore, posizionare il cursore su un punto libero dello schermo e premere due volte il comando "linee".
Due parole ora per commentare un poco il programma.
Per ottenere il lampeggiamento del cursore si è fatto ricorso all'istruzione *BPLOT*, che permette di costruire figure a punti. Una figura tracciata con *BPLOT* viene cancellata quando se ne ritraccia un'altra identica proprio sopra. Per evitare che il cursore rimanga acceso accidentalmente quando si preme un tasto speciale, si sono scritte tutte le istruzioni relative all'accensione e allo spegnimento su una sola riga, che viene così completata prima di effettuare il salto.
Per quanto riguarda il tracciamento delle figure, indubbiamente quello di linee o quadrilateri non comporta grosse difficoltà e uno sguardo alle istruzioni fugherà qualsiasi dubbio. Più difficile il tracciamento dell'ellisse, dei semicerchi e dei quarti di cerchio, per i quali si sono utilizzate le istruzioni *IDRAW* (tracciamento incrementale) e *IMOVE* (spostamento incrementale).
L'ellisse viene tracciata per quarti di 90° (ad incremento di 22°,5) per permettere anche la creazione dei semicerchi e dei quarti.
Al fine di rendere il programma più chiaro, per le variabili si è fatto uso di nomi che ricordassero la loro funzione: *C$* è la stringa per il tracciamento del cursore con *BPLOT*, *S$* è il disegno della freccetta a sinistra, *B$* è il disegno della freccetta verso il basso, *R* è il "flag" che fa comparire le crocette di riferimento se attivato, *O* è la coordinata orizzontale del cursore, *V* è la coordinata verticale dello stesso, *P* è il numero d'ordine del punto immesso, *O1* e *V1* sono le coordinate del *primo* punto immesso, *O2* e *V2* quelle del secondo punto, *Q* il "flag" per i quarti, *E* la variabile che indica le varie opzioni per il tracciamento dell'ellisse e infine *T$* è il testo che deve essere scritto sul grafico.

Figura 2 - Il cerchio ABC è stato tracciato impostando A e poi B; quello ABD è stato tracciato impostando B e poi A.

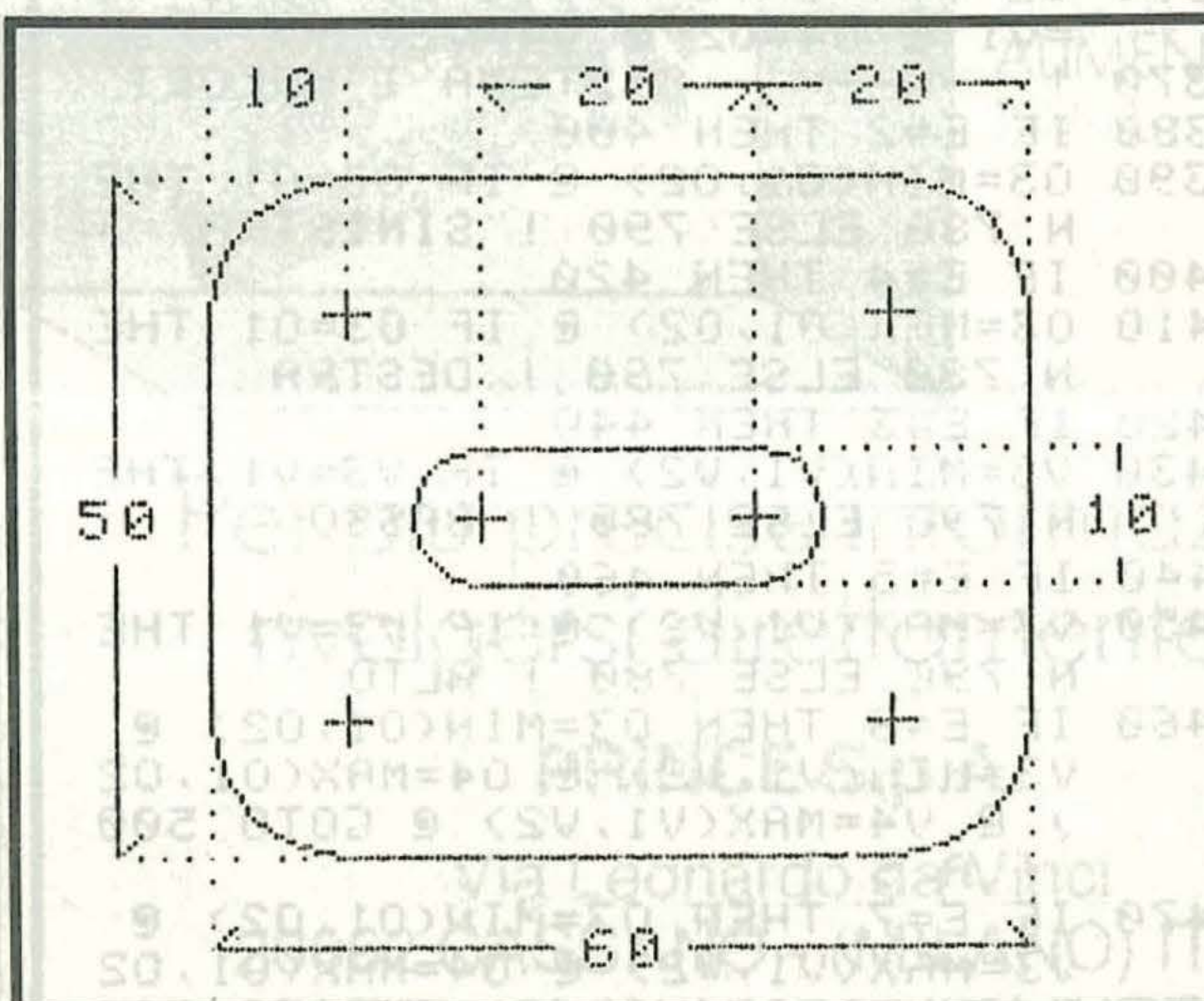
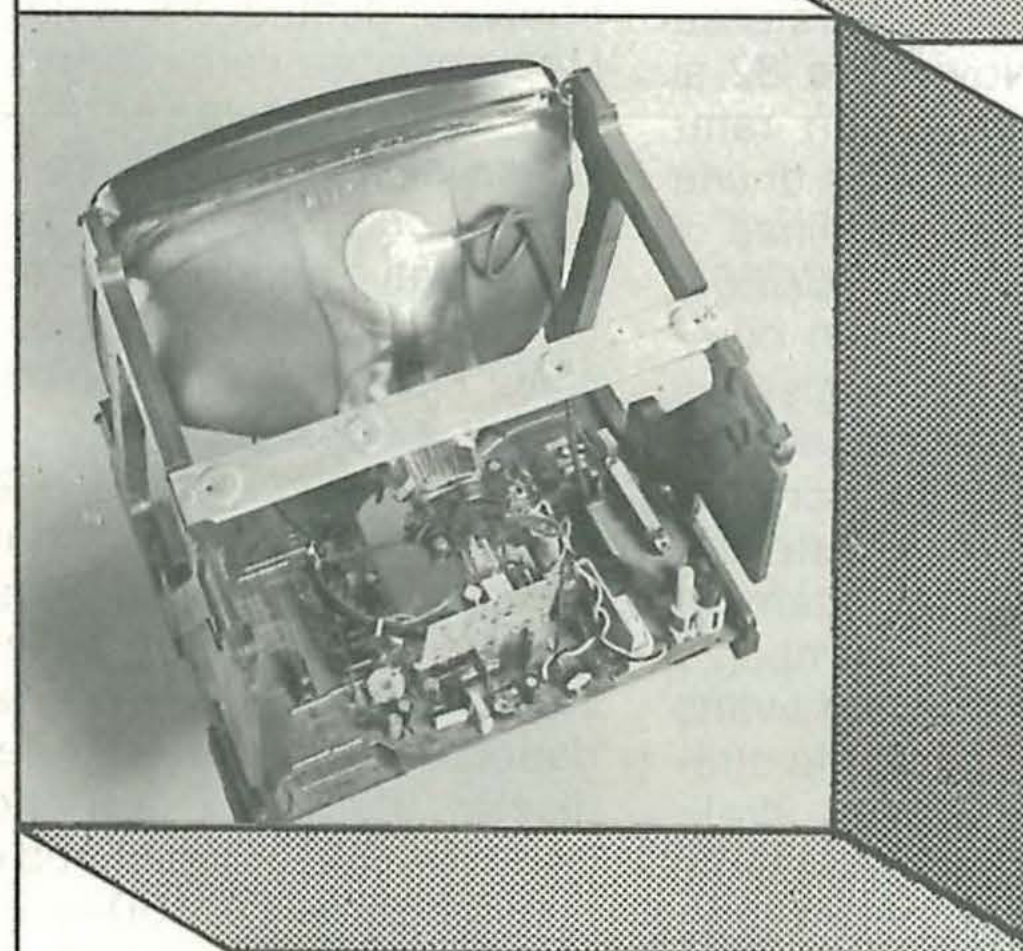
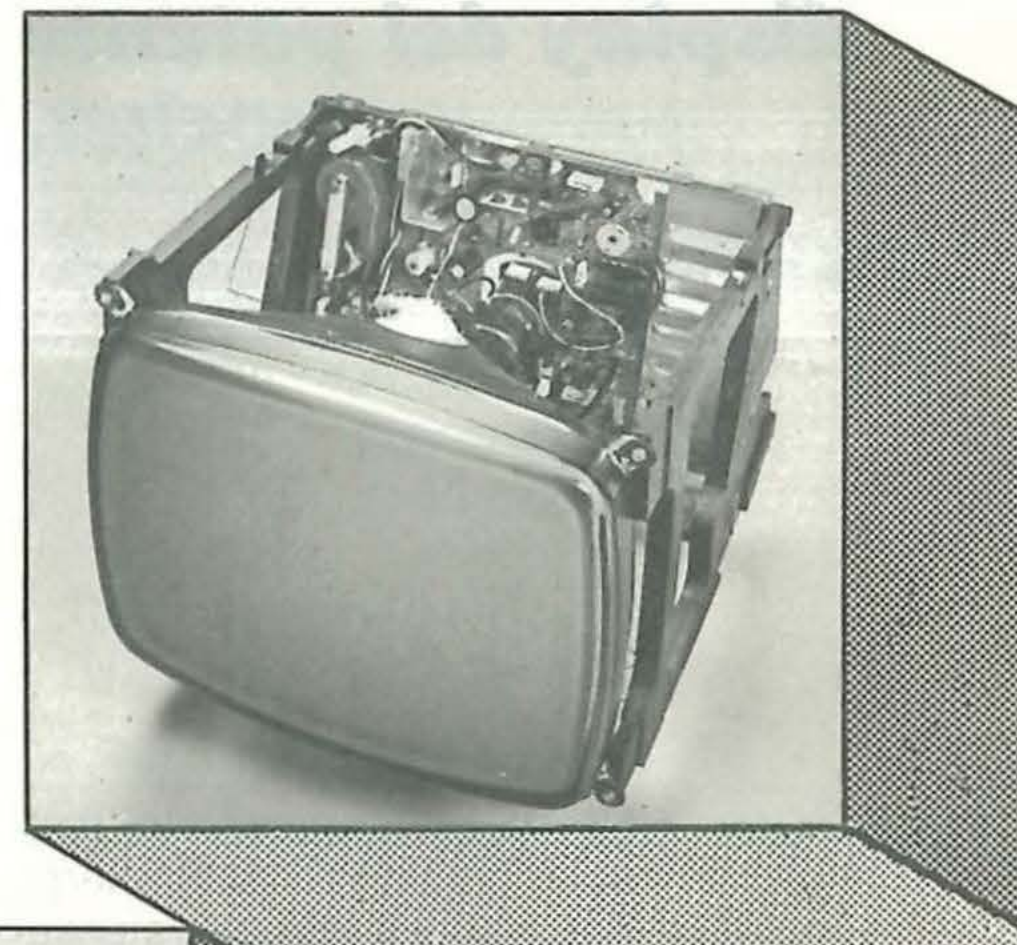



Figura 3 - Disegno quotato.

Seguito listato disegno tecnico per HP85

```
 10 ! * GRAF85 *
 20 DIM T$[31]
 30 SCALE 0,255,0,191 @ DEG
 40 C$=CHR$(0)&CHR$(0)&CHR$(0)&C
    HR$(20)&CHR$(8)&CHR$(20) @ S
    $=" @"&CHR$(248)&"@ "
 50 B$="     "&CHR$(168)&CHR$(112
    )&" "
 60 CLEAR @ DISP "RIFERIMENTI? (
    SI/NO)" @ INPUT T$@ IF UPC$(
    T$[1,1])="N" THEN R=0 ELSE R
    =1
 70 ! -------------------------
 80 GCLEAR @ IF R THEN GOSUB 131
    0
 90 O=124 @ V=112 @ P=0 @ E=1
100 XAXIS 24 @ MOVE 8,12 @ LABEL
     "ELLISS" @ MOVE 72,12 @ LAB
    EL "RETTAN"
110 MOVE 136,12 @ LABEL "LINEE"
    @ MOVE 192,12 @ LABEL "COMAN
    DI"
120 MOVE 28,5 @ BPLOT S$,1 @ MOV
    E 92,0 @ LABEL CHR$(10)
130 MOVE 152,7 @ BPLOT B$,1 @ MO
    VE 216,0 @ LABEL CHR$(125)
140 ! -------------------------
150 ON KEY# 1,"SINIST" GOTO 260
160 ON KEY# 2,"ALTO" GOTO 280
170 ON KEY# 3,"BASSO" GOTO 300
180 ON KEY# 4,"DESTRA" GOTO 320
190 ON KEY# 5,"ELLISS" GOTO 340
200 ON KEY# 6,"RETTAN" GOTO 800
210 ON KEY# 7,"LINEE" GOTO 860
220 ON KEY# 8,"COMANDI" GOTO 960
230 ! ---------------- CURSORE
240 MOVE O,V @ BPLOT C$,1 @ MOVE
     O,V @ BPLOT C$,1 @ GOTO 240
250 ! ----- SPOSTAMENTO CURSORE
260 O=O-4 @ IF O<0 THEN O=0
270 GOTO 230
280 V=V+4 @ IF V>188 THEN V=188
290 GOTO 230
300 V=V-4 @IF V<36 THEN V=36
310 GOTO 230
320 O=O+4 @ IF O>248 THEN O=248
330 GOTO 230
340 ! ----------------- ELLISSI
350 P=P+1 @ ON P GOTO 950,360
360 O2=O+4 @ V2=V-4 @ O3=O1 @ V3
    =V1 @ O4=O2 @ V4=V2
370 ! -------- SISTEMA I VALORI
380 IF E#2 THEN 400
390 O3=MIN(O1,O2) @ IF O3=O1 THE
    N 780 ELSE 790 ! SINISTRA
400 IF E#4 THEN 420
410 O3=MAX(O1,O2) @ IF O3=O1 THE
    N 780 ELSE 790 ! DESTRA
420 IF E#3 THEN 440
430 V3=MIN(V1,V2) @ IF V3=V1 THE
    N 790 ELSE 780 ! BASSO
440 IF E#5 THEN 460
450 V3=MAX(V1,V2) @ IF V3=V1 THE
    N 790 ELSE 780 ! ALTO
460 IF E=6 THEN O3=MIN(O1,O2) @
    V3=MIN(V1,V2) @ O4=MAX(O1,O2
    ) @ V4=MAX(V1,V2) @ GOTO 500
     ! A/S
470 IF E=7 THEN O3=MIN(O1,O2) @
    V3=MAX(V1,V2) @ O4=MAX(O1,O2
    ) @ V4=MIN(V1,V2) @ GOTO 500
     ! B/S
480 IF E=8 THEN O3=MAX(O1,O2) @
    V3=MAX(V1,V2) @ O4=MIN(O1,O2
    ) @ V4=MIN(V1,V2) @ GOTO 500
     ! B/D
490 IF E=9 THEN O3=MAX(O1,O2) @
    V3=MIN(V1,V2) @ O4=MIN(O1,O2
    ) @ V4=MAX(V1,V2) ! A/D
500 ! ---
510 O2=O3-O4 @ O1=MAX(O3,O4) @ I
    F O1=O4 THEN O1=O3-O2*2 @ O2
    =O4 ELSE O1=O3 @ O2=O4
520 V2=V3-V4 @ V1=MIN(V3,V4) @ I
    F V1=V4 THEN V2=V4+V2*2 @ V1
    =V3 ELSE V1=V3 @ V2=V4
530 MOVE O1,V1
540 ! -- SELEZIONA I QUADRANTI
550 FOR I=22.5 TO 90 STEP 22.5
560 IF E=1 OR E=4 OR E=5 OR E=9
    THEN 580
570 GOSUB 740 @ NEXT I @ GOTO 59
    0
580 GOSUB 730 @ NEXT I
590 FOR I=112.5 TO 180 STEP 22.5
600 IF E=1 OR E=2 OR E=5 OR E=6
    THEN 620
610 GOSUB 740 @ NEXT I @ GOTO 63
    0
620 GOSUB 730 @ NEXT I
630 FOR I=202.5 TO 270 STEP 22.5
640 IF E=1 OR E=2 OR E=3 OR E=7
    THEN 660
650 GOSUB 740 @ NEXT I @ GOTO 67
    0
660 GOSUB 730 @ NEXT I
670 FOR I=292.5 TO 360 STEP 22.5
680 IF E=1 OR E=3 OR E=4 OR E=8
    THEN 700
690 GOSUB 740 @ NEXT I @ GOTO 71
    0
700 GOSUB 730 @ NEXT I
710 E=1 @ GOTO 750
720 ! ----- TRACCIA LE ELLISSI
730 IDRAW (COS(I-22.5)-COS(I))*(
    O2-O1),-((SIN(I-22.5)-SIN(I)
    )*ABS(V2-V1)) @ RETURN
740 IMOVE (COS(I-22.5)-COS(I))*(
    O2-O1),-((SIN(I-22.5)-SIN(I)
    )*ABS(V2-V1)) @ RETURN
750 ! ---
760 P=0 @ PEN 1 @ IF R THEN GOSU
    B 1310
770 GOTO 230
780 O3=O1 @ O4=O2 @ V3=V1 @ V4=V
    2 @ GOTO 500
790 O3=O2 @ O4=O1 @ V3=V2 @ V4=V
    1 @ GOTO 500
800 ! --------------- RETTANGOLI
810 P=P+1 @ ON P GOTO 950,820
820 O2=O+4 @ V2=V-4
830 MOVE O2,V2 @ DRAW O1,V2 @ DR
    AW O1,V1@ DRAW O2,V1 @ DRAW
     O2,V2
840 P=0 @ PEN 1 @ IF R THEN GOSU
    B 1310
850 GOTO 230
860 ! ------------------- LINEE
870 P=P+1 @ ON P GOTO 950,880
880 O2=O+4 @ V2=V-4 @ P=0
890 IF NOT (V1=V2 AND V1=32) THE
    N 930
900 IF NOT (O1=4 AND O2=252) THE
    N 930
910 XAXIS V1 @ PEN 1 @ IF R THEN
     GOSUB 1310
```

```
920 GOTO 230
930 MOVE O2,V2 @ DRAW O1,V1 @ PE
    N 1 @ IF R THEN GOSUB 1310
940 GOTO 230
950 PENUP @ O1=O+4 @ V1=V-4 @ PL
    OT O1,V1 @ PENUP @ GOTO 230
960 ! ----------------- COMANDI
970 CLEAR @ Q=0 @ DISP "
       * COMANDI *" @ DISP
980 DISP "TESTO: per scrivere su
    l grafico"
990 DISP "COPIA: per copiare il
    grafico           sulla stamp
    ante"
1000 DISP "QUARTI:per passare da
      semicerchi        a quarti
     di cerchio"
1010 DISP "CANCELLA: per predisp
     orre alla          cancellaz
     ione"
1020 DISP "S A/S, A B/S, B B/D,
     D A/D: per         scegliere
      il quadrante" @ DISP
1030 ON KEY# 1,"S A/S" GOTO 1130
1040 ON KEY# 2,"A B/S" GOTO 1150
1050 ON KEY# 3,"B B/D" GOTO 1170
1060 ON KEY# 4,"D A/D" GOTO 1190
1070 ON KEY# 5,"TESTO" GOTO 1210
1080 ON KEY# 6,"COPIA" GOTO 1250
1090 ON KEY# 7,"QUARTI" GOTO 127
     0
1100 ON KEY# 8,"CANCELLA" GOTO 1
     290
1110 KEY LABEL
1120 GOTO 1120
1130 IF Q THEN E=6 ELSE E=2
1140 GOTO 140
1150 IF Q THEN E=7 ELSE E=5
1160 GOTO 140
1170 IF Q THEN E=8 ELSE E=3
1180 GOTO 140
1190 IF Q THEN E=9 ELSE E=4
1200 GOTO 140
1210 ! ------------------- TESTO
1220 CLEAR @ DISP "SCRIVERE IL T
     ESTO FRA VIRGOLETTE" @ INPU
     T T$
1230 MOVE O+2,V-8 @ LABEL T$ @ P
     EN 1 @ IF R THEN GOSUB 1310
1240 GOTO 140
1250 ! ------------------- COPIA
1260 GCLEAR 24 @ COPY @ GOTO 70
1270 ! ------------------ QUARTI
1280 Q=1 @ DISP "QUARTI" @ GOTO
     1120
1290 ! --------------- CANCELLA
1300 PEN -1 @ GOTO 140
1310 ! ------------- RIFERIMENTI
1320 FOR I=184 TO 32 STEP -76 @
     FOR J=4 TO 252 STEP 124 @ G
     OSUB 1330 @ NEXT J @ NEXT I
      @ RETURN
1330 MOVE J,I @ IMOVE 0,2 @ IDRA
     W 0,-4 @ IMOVE -2,2 @ IDRAW
      4,0 @ RETURN
1340 END
```

SHARP

Hard copy per PC 1500

Una interessante routine di trasferimento su carta della grafica possibile tramite il display del potente pocket.

di I. Aschieri e L. Mori

Come si ricorderà, nel test sul modello PC 1500 Sharp, apparso su queste stesse pagine nel Novembre '82, si era sottolineata con rammarico la mancanza di una istruzione (simile in linea di principio al *SAVE SCREEN* di alcuni personal in commercio) che consentisse di ottenere su carta le interessanti configurazioni creabili sul display. Le caratteristiche del PC 1500, unitamente all'interfaccia stampante CE 150, non consentivano infatti di combinare le straordinarie possibilità grafiche della stampante con le inconsuete particolarità che rendono il display un vero e proprio schermo (seppure di limitate dimensioni).

La procedura che presentiamc è di notevole interesse per essere riusciti a far interagire in modo diretto due unità di output aventi peculiarità molto dissimili. Inoltre in tale procedura il display viene utilizzato sia come output che come "finto" input.

Questa utility ha lo scopo di riprodurre su carta ciò che, in un modo o nell'altro, l'utente è riuscito a portare sul display. In prima istanza si è pensato di realizzare una hard-copy imitativa del display, vale a dire che il disegno che compare sul display viene riprodotto attraverso una serie di box quadrati (equivalenti ai dot) ottenuti mediante l'opzione B dello statement *LINE*.

La copia su carta è stata orientata nel senso della lunghezza al fine di poter usufruire di una maggiore elasticità nelle dimensioni dei dot. Nelle sue linee generali, la procedura effettua uno scanning colonna per colonna del display e disegno su carta solo i quadrati corrispondenti ai dot settati. Una possibile variante (non analizzata in questa sede) è quella che prevede l'uso di due colori contrastanti per i dot settati e per quelli resettati: in tal caso sarebbe possibile riprodurre l'intero reticolo 7 x 156 del display.

Siamo infine convinti che la procedura proposta consenta di intravvedere una fitta schiera di future utilizzazioni, sia connesse con la rispettabile risoluzione del display (ben 1092 dot) che con le ormai indiscutibili possibilità del CE 150.

Ma vediamo un po' più da vicino la procedura.

I	*Variabile contatore.*
G	*Dimensione del dot di stampa (numero di punti).*
K	*Variabile di shift orizzontale (ogni ciclo è decrementata di G+1).*
Y	*Contiene gli esponenti da assegnare a 2 ai fini della comparazione AND (riga 60).*

Figura 1 - Variabili impiegate.

Riga 10

La label alfabetica consente di inserire l'utility in un qualunque programma-utente che, nelle fasi più opportune, può richiamarla semplicemente facendo riferimento alla label stessa, ad esempio:

```
100  INPUT "VUOI IL
     PLOTTING?"; A$
110  IF A$ = "SI" GOSUB
     "PLOT"
```

La riga contiene inoltre un loop che provvede al salvataggio colonna per colonna del contenuto del display. Si noti che le locazioni di salvataggio sono quelle immediatamente seguenti l'ultima locazione di memoria occupata dall'insieme del programma-utente e della subroutine di plotaggio. Si tratta quindi di una allocazione dinamica dei dati da elaborare (*STATUS* 2 fornisce l'indirizzo della prima locazione libera dopo che il programma-utente e la subroutine sono stati immagazzinati in RAM).

Riga 20

La label alfabetica consente il ritorno automatico a questa medesima riga nel caso di input di dati errati. Ciò rende l'intera routine completamente slegata dalle label numeriche e perciò può essere completamente riallocata in una qualsiasi zona di programma.

La riga stabilisce innanzitutto la durata del *PRINT*ing sul display (circa 0.8 secondi), il che consente di non interrompere l'esecuzione del programma in corrispondenza degli statement *PRINT* (come avverrebbe con l'opzione-default *WAIT*).

La richiesta di input è la dimensione (numero di punti) dei quadrati-dot che compariranno su carta: il minimo di 2 costituisce il più piccolo quadratino (al di sotto si avrebbe un solo punto), mentre il 28 rappresenta il massimo quadrato-dot tracciabile; tale valore viene ottenuto tenendo conto del numero dei dot (7), della massima escursione sulle ascisse (220 punti) e dell'interspazio fra punto e punto (1).

Riga 30

Viene prima di tutto cancellato il display, quindi vengono recuperati i dati salvati in precedenza e sottoposti allo statement *GPRINT*. In tal modo sul display ricomparirà esattamente quanto vi si trovava prima dell'*INPUT* della riga *20*. In questa fase si può osservare una certa lentezza nell'operazione di recupero e scrittura.

Riga 40

Viene selezionata l'opzione *GRAPH* al fine di poter utilizzare successivamente lo statement *LINE*.

Successivamente si procede alla centratura della testina di stampa, affinchè il risultato (plotting) venga a trovarsi esattamente al centro della banda cartacea. In tale posizione, infine, viene fissata la sorgente cartesiana delle coordinate di stampa.

Righe 50-70

Viene riconosciuto lo stato di attivazione dei 7 dot colonna per colonna.

Righe 60-70

Si confronta il valore fornito dalla funzione *POINT* con le potenze successive del 2 (da 2^0 a 2^6). Se la comparazione *AND* ha un esito positivo viene tracciato il corrispondente dot.

A questo punto si possono osservare due fatti:

a) è abbastanza inusuale l'impiego della comparazione *AND* della riga *60*. Normalmente i campi

Figura 2 - Un esempio di plottaggio.

numerici coinvolti nelle condizioni di branching vengono confrontati in senso numerico e non in senso binario, come in questo caso. Sarebbe lungo, e forse fuori luogo, indagare sul fatto, ma possiamo dire che una tale possibilità è stata riscontrata solo su pochissimi sistemi, comunque assai più evoluti. Infine il lettore è invitato a riflettere sulle seguenti routine:

```
10  INPUT A: IF A<0 OR A>
    127 THEN 10
20  PRINT A; "=";
30  FOR I=0 TO 6: B=2^I
40  IF B AND A THEN PRINT
    "1"
50  GOTO 70
60  PRINT "0"
70  NEXT I: GOTO 10
```

b) dopo aver tracciato tutti i dot settati di una colonna, è necessario cambiare la posizione dell'origine in senso verticale, poichè il CE 150 presenta la limitazione grafica da —2047 a +2047 punti sulle ordinate. Se non si operasse così si rischierebbe di uscire da tali limiti interrompendo il plotting.

Riga 80

Viene restituito il controllo al main-program ripristinando preventivamente le condizioni preesistenti di stampa su display e su stampante.

Figura 3
Il listato per PC 1500.

```
 1:REM  ********
  ** PLOTTER ***
  *******
  
  *
10:"PLOT":FOR I=0
   TO 155:POKE ((
   STATUS 2)+I),
   POINT I:NEXT I
20:"DIME":WAIT 0:
   INPUT "DIMENSI
   ONE DOT (2-28)
   ?";G: IF G<2OR
   G>28GOTO "DIME
   "
30:CLS :FOR I=0TO
   155:GCURSOR I:
   GPRINT (PEEK (
   (STATUS 2)+I))
   ;NEXT I
40:GRAPH :
   GLCURSOR (110+
   ((G*7)/2),-50)
   ;SORGN
50:FOR I=0TO 155:
   K=0:GLCURSOR
   0,-G):SORGN
60:FOR Y=0TO 6:H=
   2^Y: IF HAND
   POINT ILINE (K
   ,-G)-(K-G,-2*G
   ),,2,B
70:K=K-G-1:NEXT Y
   ;NEXT I
80:WAIT :TEXT :
   RETURN
```

sinclair ZX81

Atterraggio strumentale

Come pilotare un Jumbo stando comodamente seduti in poltrona.

Linate ore 9 del mattino. Una fitta nebbia grava sulla pista. Il volo AZ 135 da Francoforte è a corto di carburante, e deve atterrare. Chiede alla torre di controllo l'assistenza per un atterraggio strumentale.

I controllori di volo inquadrano l'aereo sul radar quando è a circa 30 Km. di distanza; danno quindi al pilota le istruzioni necessarie per atterrare.

Atterraggio strumentale" è un classico gioco di simulazione.

Il programma inizia enunciando la situazione di partenza e le regole. L'aereo dispone di 1.500 lt. di carburante e deve cercare di economizzarlo al massimo. Deve arrivare sulla pista a non più di 250 Km / h e d'altronde non può volare al di sotto di tale velocità.

Le variazioni di rotta non possono essere più di 10⁸ ogni 15 secondi, cioè ad ogni nuova direttiva della torre di controllo.

Infine, se vola molto basso lontano dalla pista, è probabile che incappi in qualche ostacolo, sfracellandosi.

Il programma fa quindi vedere lo schermo radar della torre di controllo, nel momento in cui vi entra l'aereo, stampa le condizioni iniziali e chiede le decisioni.

Queste vertono su:

- Rotta;
- Posizione degli alettoni (da + 3 a — 3, per salire o scendere di quota);
- Posizione manetta del gas (da 0 a 10).

Il gioco consiste nel portare l'aereo sulla pista ad una velocità abbastanza bassa da poter atterrare, ma abbastanza alta da non andare in stallo, e ad una quota tale che con un ultimo spostamento degli alettoni si possa toccare terra.

Se si arriva sulla pista troppo veloci o troppo alti, non resta che riprendere quota, fare un altro giro e ritentare, sempre che le scorte di carburante lo permettano.

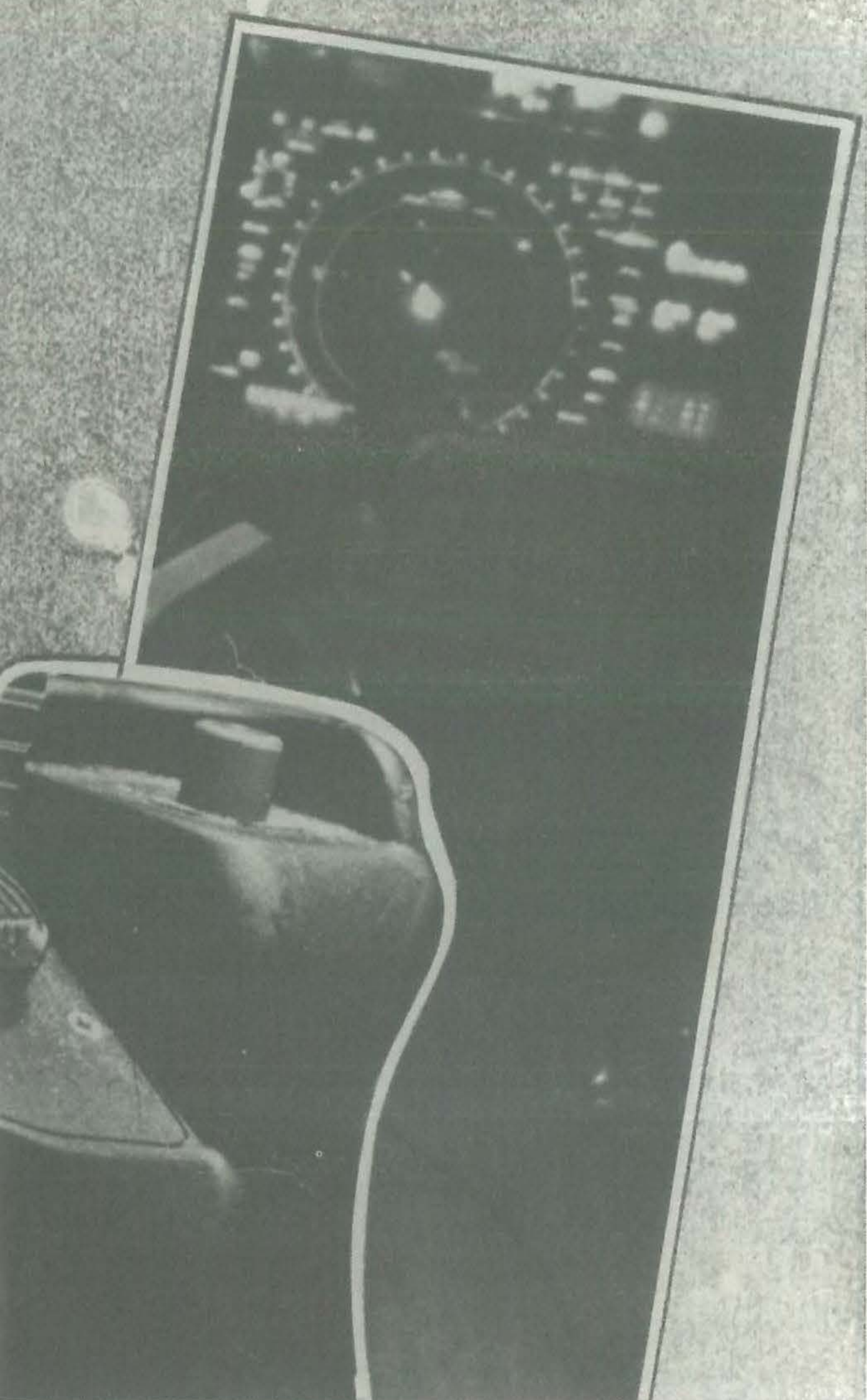
Dopo ogni serie di decisioni il programma visualizza la nuova posizione dell'aereo sullo schermo radar e la nuova situazione.

Sui calcoli c'è poco da dire. Visto che la situazione viene aggiornata ogni 15 secondi, dalla velocità si determina la ditanza percorsa in quell'intervallo; da tale distanza e dalla rotta (angolo di direzione rispetto al Nord) si arriva, con semplici formule trigonometriche, alla nuova posizione, che viene plottata sullo schermo radar. La distanza dalla torre di controllo viene invece calcolata con il teorema di Pitagora.

Si è introdotto un fattore di casualità che provoca la caduta dell'aereo, per scontro con un ostacolo, con 1 probabilità su 4 se l'aereo viaggia lontano dalla pista a quota di 50 mt., e con probabilità maggiore a quote inferiori.

Le variazioni di altezza si sono fatte, evidentemente, proporzionali alla posizione degli alettoni ed anche alla velocità.

10-410	*Enuncia le regole del gioco.*
415-467	*Inizializza le variabili.*
470	*Manda alla subroutine che visualizza lo schermo radar.*
480	*Stampa la posizione iniziale dell'aereo.*
500-580	*Stampa la situazione.*
590-680	*Riceve le istruzioni.*
690-760	*Calcola la nuova situazione in base alle istruzioni ricevute.*
810-915	*Serie di IF che stabiliscono se il gioco è finito, o con l'atterraggio o con qualche disastro.*
935-950	*Manda alla subroutine che visualizza lo schermo radar e stampa la nuova posizione dell'aereo.*
1000	*Rimanda a 500 se il gioco non è finito.*
1100-1430	*Stampa i vari messaggi di chiusura a secondo delle varie conclusioni del gioco.*
1500-1540	*Rinvia all'inizio se si vuole continuare a giocare.*
2000-2100	*Subroutine di stampa dello schermo radar.*

Figura 1 - Note relative al programma.

Il consumo di carburante è proporzionale alla posizione della manetta del gas, fermo restando un minimo che si consuma anche a gas 0. La velocità varia, oltre che in funzione del gas, anche della posizione degli alettoni, perchè è chiaro che l'aereo in picchiata acquista velocità e in cabrata rallenta.

In caso di disperata scarsezza di carburante, si potrà quindi mantenere un minimo di velocità scendendo opportunamente di quota.

H	*Altezza dell'aereo. (mt)*
D	*Distanza dalla pista. (Km)*
F	*Quantità di carburante. (lt)*
V	*Velocità. (Km/h)*
R	*Rotta. (in gradi)*
A	*Posizione degli alettoni. (-3 ÷ + 3)*
G	*Posizione manetta del gas. (0 ÷ 10)*
X—Y	*Coordinate dell'aereo sullo schermo.radar.*
A$	*Variabile di servizio, il cui input serve solo a bloccare l'elaborazione e a riprenderla.*
S	*Spazio percorso dall'aereo nell'intervallo fra una serie di istruzioni e la successiva.*

Figura 2 - Elenco delle variabili:

```
  10 PRINT "      ATTERRAGGIO STR
UMENTALE"
  20 PRINT
  30 PRINT
  40 PRINT "VUOI LA STRINGA INFO
RMATIVA?"
  50 INPUT A$
  60 IF A$="NO" THEN GOTO 300
  70 CLS
  80 PRINT "IL VOLO AZ 135 DA FR
ANCOFORTE"
  90 PRINT "DEVE ATTERRARE SU LI
NATE."
 100 PRINT "SULL""AEROPORTO GRAV
A UNA FITTA"
 110 PRINT "NEBBIA;LA VISIBILITA
""E""RIDOTTA"
 120 PRINT "A POCHI METRI;L""ATT
ERRAGGIO"
 130 PRINT "VERRA""QUINDI GUIDAT
O DA TERRA"
 140 PRINT "A MEZZO RADAR."
 145 PRINT "VOI SIETE I CONTROLL
ORI DI VOLO"
 150 PRINT "E DOVRETE DARE AI PI
LOTI LE"
 160 PRINT "ISTRUZIONI NECESSARI
E PER ATTER-"
 170 PRINT "RARE."
 180 PRINT "TENETE PRESENTE CHE:
"
 190 PRINT "BISOGNA ARRIVARE SUL
LA PISTA A"
 200 PRINT "VELOCITA""INFERIORE
A 250 KM/H"
 210 PRINT "E CHE NON SONO POSSI
BILI"
 220 PRINT "CAMBIAMENTI DI ROTTA
 DI PIU""DI"
 230 PRINT "10 GRADI AD OGNI INT
ERVALLO DI"
 240 PRINT "15 SEC.I CONTROLLORI
 DI VOLO"
 250 PRINT "DARANNO NUOVE ISTRUZ
IONI E LEG-"
 260 PRINT "GERANNO I DATI SUL R
ADAR OGNI"
 270 PRINT "15 SECONDI."
 280 LET B$="PREMI UN TASTO PER
CONTINUARE"
 290 PRINT B$
 295 INPUT A$
 300 CLS
 310 PRINT "L""AEREO COMPARE SUL
LO SCHERMO"
 320 PRINT "A CIRCA 30 KM DALLA
PISTA;"
 340 PRINT "VA ALLA VELOCITA""DI
 550 KM/H"
 350 PRINT "ED E"" ALL""ALTEZZA
DI 3000 MT."
 360 PRINT "DISPONE DI 1500 LT D
I CARBURANTE"
 370 PRINT "E HA ROTTA 180 GRADI
"
 380 PRINT "LA MANETTA DEL GAS E
""IN POSI-"
 390 PRINT "ZIONE 1 (0/10) E GLI
 ALETTONI"
 400 PRINT "SONO A -1 (-3/+3)."
 410 PRINT AT 20,0;"PREMI UN TAS
TO PER CONTINUARE"
 412 INPUT A$
 415 LET D=30
 420 LET V=550
 430 LET H=3000
 440 LET R=180
 450 LET A=-1
 455 LET Y=RND*22+22
 460 LET X=31+31*(1-2*INT (RND*2
))+RND*2
 465 LET G=1
 467 LET F=1500
 470 GOSUB 2000
 480 PLOT X,Y
 490 INPUT A$
 500 CLS
 505 PRINT "-----------SITUAZION
E-----------"
 507 PRINT
 510 PRINT "DISTANZA:";D;" KM"
 520 PRINT "VELOCITA"":";V;" KM/
H"
```

Figura 3 - Il listato del programma Atterraggio strumentale.

Atterraggio strumentale

```
 530 PRINT "ALTEZZA:";H;" MT"
 540 PRINT "ROTTA:";R;" GRADI"
 550 PRINT "POSIZ.ALETTONI:";A
 560 PRINT "GAS:";G
 570 PRINT "CARBURANTE:";F
 580 PRINT
 590 PRINT
 595 LET T=R
 600 PRINT "NUOVA ROTTA?";
 610 INPUT R
 613 IF R>=350 OR R<=10 THEN GOT
O 620
 615 IF ABS (T-R)>10 THEN GOTO 6
10
 620 PRINT R
 630 PRINT "NUOVA POSIZIONE ALET
TONI?(-3/+3)";
 640 INPUT A
 650 PRINT A
 660 PRINT "NUOVA POSIZ.MANETTA
GAS?(0/10)";
 670 INPUT G
 680 PRINT G
 690 LET H=H+A*(V/3)
 700 LET F=F-30-G*10
 710 LET V=V-50+G*20-15*A
 720 LET S=((V/3.6)*15)/1000
 730 LET X=X+S*COS (PI/2-R/57.32
)
 740 LET Y=Y+S*SIN (PI/2-R/57.32
)
 750 LET DQ=(ABS (X-31))**2+(ABS
 (Y-21))**2
 760 LET D=SQR (ABS DQ)
 800 IF H<=25 AND D<=1 AND V<=25
0 THEN GOTO 1100
 810 IF (INT (RND*4)+50/H)>=5 AN
D D>2 THEN GOTO 1400
 815 IF H<0 AND V>250 THEN GOTO
1400
 900 IF F<=0 THEN GOTO 1200
 905 IF H<5 AND D>1 THEN GOTO 14
00
 909 IF V<250 AND D>1 THEN GOTO
1300
 915 IF V>1200 THEN LET V=1200
 935 GOSUB 2000
 940 PLOT X,Y
 950 INPUT A$
1000 GOTO 500
```

```
1100 CLS
1110 PRINT AT 10,5;"ATTERRAGGIO
RIUSCITO"
1120 PRINT AT 12,10;"COMPLIMENTI
"
1130 GOTO 1500
1200 CLS
1210 PRINT AT 10,5;"L""AEREO E""
PRECIPITATO"
1220 PRINT AT 12,5;"IL CARBURANT
E E""FINITO"
1230 GOTO 1500
1300 CLS
1310 PRINT AT 10,5;"L""AEREO E""
PRECIPITATO"
1320 PRINT AT 12,0;"E""ANDATO IN
 STALLO PER SCARSA VELOCITA"""
1330 GOTO 1500
1400 CLS
1410 PRINT AT 10,5;"L""AEREO E""
PRECIPITATO"
1420 PRINT AT 12,0;"SI E"" SFRAC
ELLATO CONTRO UN OSTACOLO"
1430 GOTO 1500
1500 INPUT A$
1505 CLS
1510 PRINT "          VUOI RIPROVARE
?"
1520 INPUT A$
1530 IF A$="SI" THEN GOTO 300
1540 IF A$="NO" THEN STOP
2000 CLS
2005 FOR K=29 TO 34
2010 PLOT K,21
2020 PLOT K,22
2030 NEXT K
2040 PRINT AT 0,0;"330";AT 0,15;
"0";AT 0,29;"30"
2050 PRINT AT 5,0;"300";AT 5,29;
"60"
2060 PRINT AT 10,0;"270";AT 10,2
9;"90"
2070 PRINT AT 15,0;"240";AT 15,2
9;"120"
2080 PRINT AT 21,0;"210";AT 21,2
9;"150"
2085 PRINT AT 21,14;"180"
2090 RETURN
```

Usare il sistema operativo CP/M

IL LIBRO

Il sistema operativo CP/M è stato progettato per rendere semplice l'uso di un microcomputer. Questo libro vi renderà semplice l'uso del CP/M. (Le versioni esaminate del CP/M sono il CP/M 1.4-il CP/M 2.2. e il nuovo sistema operativo multiutente MP/M) La maggior parte di utenti di microcomputer dovrà, infatti, un giorno o l'altro, fare ricorso al CP/M, disponibile su quasi tutti i computer basati sui microprocessori 8080 e Z80, come pure su certi sistemi utilizzanti il 6502. Il libro, senza presupporre alcuna conoscenza di un calcolatore, inizia con la descrizione, passo-passo delle procedure di inizializzazione del sistema: accensione, inserimento dei dischetti, esecuzione delle più comuni operazioni su file, compresa la duplicazione dei dischetti. Prosegue con il PIP (programma di trasferimento dei file), il DDT (programma di messa a punto) e ED (programma editor). Per entrare sempre più, fornendo numerosi consigli pratici, all'interno del CP/M e delle sue operazioni, al fine di comprenderne appieno le risorse ed eventualmente dare gli strumenti per successive modifiche.

Pagg. 320 Cod. 510P
L. 22.000
(Abb. L. 19.800)

GRUPPO EDITORIALE JACKSON
Divisione Libri

Tagliando d'ordine da inviare a:
Gruppo Editoriale Jackson - Via Rosellini, 12 - 20124 Milano

NOME ______________

COGNOME ______________

INDIRIZZO ______________

CAP ________ CITTA' ______________

Partita IVA (indispensabile per le aziende) ______________

☐ Inviatemi il **Libro CP/M**

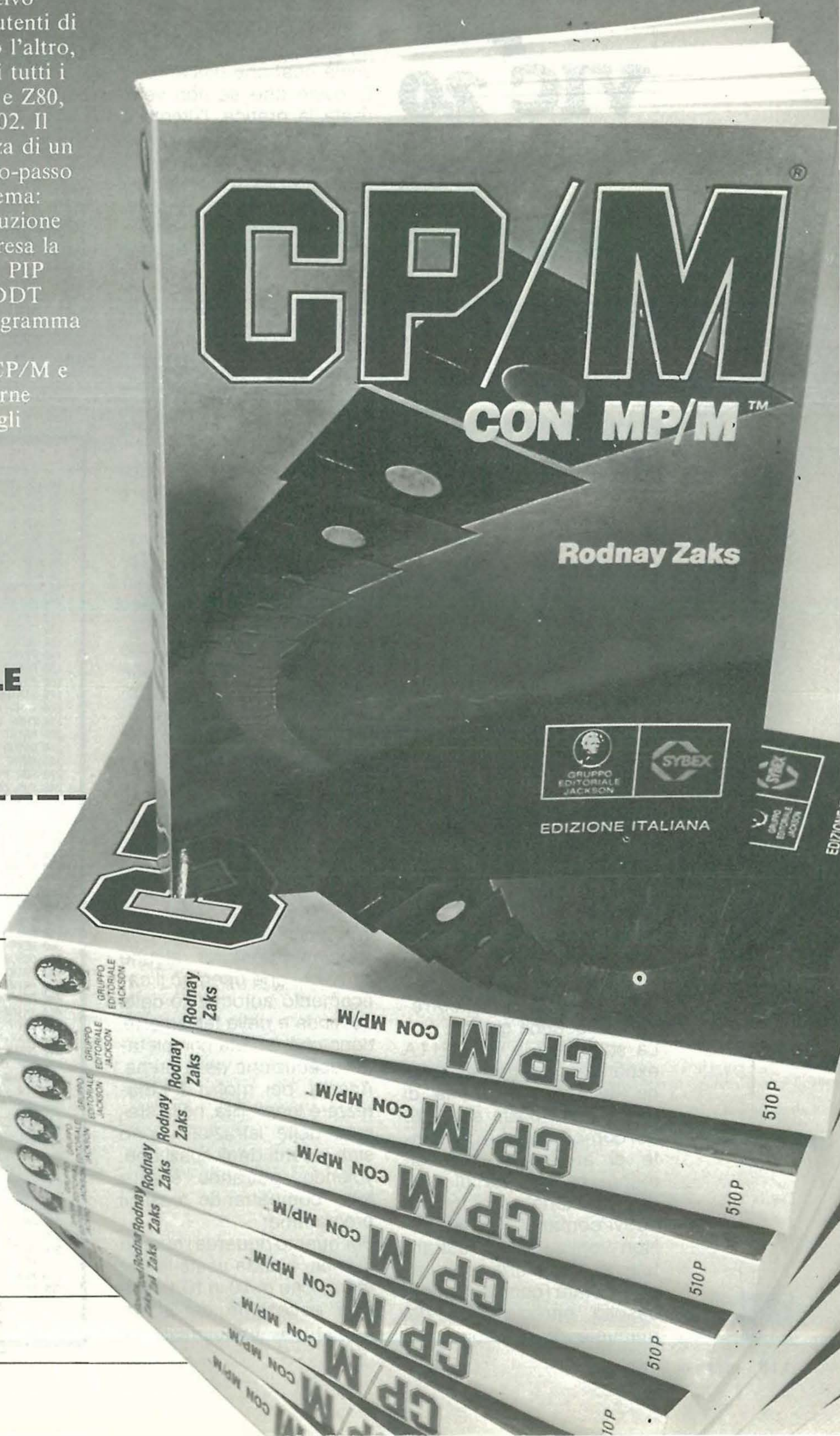
☐ Pagherò al postino l'importo indicato più L. 2.000 per le spese di spedizione.

☐ Allego assegno n° ______________

di L. ______________

☐ Non Abbonato

☐ Abbonato (sconto 10%): ______________
(indicare a quali riviste)

DATA ______________ FIRMA ______________

VIC-20

Asta che cade per VIC 20 con super expander

Un programma di didattica per il VIC 20, convertito da un programma Apple

di G. Rittarelli

Scopo principale di questo articolo, dedicato alla didattica, è quello di mostrare come molti programmi grafici scritti per Apple II possono essere abbastanza facilmente trasferiti su di un computer di caratteristiche alquanto diverse come il VIC 20, ovviamente corredato di scheda grafica.
La scheda è la VIC 1211A expander cartdrige, che viene inserita nella porta di espansione posta sul retro del computer e che consente di aumentare notevolmente la potenza di programmazione, aggiungendo nuovi comandi al BASIC.
Non vogliamo qui illustrare tutti i comandi di cui la scheda è dotata (compito questo assolto egregiamente dal manuale di istruzioni), ma diremo che colori, suoni, e in particolare modo la grafica, sono ottenuti in maniera più semplice e rapida.
Inoltre la scheda contiene un'aggiunta di 3 Kbyte di memoria RAM, che vengono tutti utilizzati per l'attivazione della pagina grafica ad alta risoluzione, e per la verità neanche bastano.
E' ovvio che se non verrà usata la grafica, l'utente si ritroverà con 3 Kbyte di memoria in più rispetto alla configurazione standard, il che gli permetterà di scrivere programmi più lunghi.
Il programma che abbiamo preso in considerazione è quello relativo all'*ASTA CHE CADE*, comparso sul numero 30 di **Bit** per la versione Apple e numero 35 per la versione Atari. Rispetto al programma originale sono state introdotte alcune variazioni.
Sono stati innanzitutto aggiunti dei titoli, per illustrare le varie fasi del programma e le istruzioni del tipo "*INPUT B$*", che avevano il solo scopo di creare una fermata nel programma, sono state sostituite con sottoprogrammi che creano una temporizzazione, con possibilità di arrestare l'esecuzione del programma premendo un tasto qualunque e di farlo ripartire allo stesso modo.
Inoltre, poichè la memoria del VIC senza ulteriori espansioni non era sufficiente ad accogliere un programma di questa lunghezza, si è suddiviso il programma stesso in tre parti, con aggiunta però di alcune istruzioni che operano il caricamento automatico della seconda e della terza parte, dopo che è stata completata l'esecuzione della prima.
Ancora, per motivi di chiarezza e leggibilità, nella stesura delle istruzioni, sono stati inseriti degli spazi che, volendo, potranno essere tolti, compattando così il programma.
Per quanto riguarda i numeri di linea, è stata usata la numerazione di 10 in 10, con la sola eccezione dei sottoprogrammi. Infatti, nelle tre parti in cui è suddiviso il programma, le linee dei sottoprogrammi, quando sono comuni, portano gli stessi numeri.
Poichè il programma fa riferimento ad un articolo pubblicato in precedenza, ci limiteremo a commentare le sole istruzioni del listato che sono tipiche del VIC. D'altra parte il programma è abbastanza semplice e si commenta da sè.
Le tre parti vanno memorizzate su cassetta, ovviamente una di seguito all'altra. Il programma gira con il VIC in configurazione standard con aggiunta della sola scheda grafica.
Per coloro che possiedono l'espansione di memoria, sarà sufficiente scrivere un programma unico, utilizzando unicamente i sottoprogrammi citati nella terza parte; le istruzioni contenute nelle linee *710* e *1340* andranno sostituite con *REM*.
Ancora un'ultima cosa. In grafica, il VIC tende a ingrossare le linee verticali. Abbiamo riscontrato che il fenomeno è legato sia al colore usato, sia al tipo di apparecchio televisivo.
Per ridurre questo inconveniente si può provare a diminuire la luminosità del televisore fino a quando le linee orizzontali e verticali avranno all'incirca lo stesso spessore. Ne risulteranno leggermente alterati i colori che perderanno un po' della loro brillantezza, ma si guadagnerà in definizione.

```
10 REM **********************
20 REM *                    *
30 REM *   ASTA CHE CADE    *
40 REM *                    *
50 REM **********************
60 COLOR 1,5,4,0: PRINT "♥QQQQQQQQQQQ  * ASTA CHE CADE *": GOSUB 3030
70 REM ========================
80 REM   I FASE - TRASLAZIONE
90 REM ========================
100 PRINT "♥QQQQQQQQQQ" TAB(7)"I  FASE": PRINT
110 PRINT TAB(5)"TRASLAZIONE": GOSUB 3030: PRINT "♥QQQQQQQQQ";
120 PRINT "    COORDINATE DI       Q UN PUNTO E DI       Q  UNA RETTA"
130 GOSUB 3030: T=150: Z=1.51: Y=1.5*T*Z: PRINT "♥QQQQQQQQQ"
140 PRINT TAB(7)"T" TAB(14)"Y": PRINT "Q"
150 PRINT TAB(5)T TAB(12)INT(Y): GOSUB 3030: GRAPHIC 3
160 POINT 1, 150,339: CHAR 8,0,"150,339": GOSUB 3030
170 X1=400: Y1=200: X2=800: Y2=800
180 DRAW 1, X1,Y1 TO X2,Y2
190 CHAR 1,4,"400,200": CHAR 18,12,"800,800": GOSUB 3040: GRAPHIC 0
200 PRINT "♥QQQQQQQQQQQ      CADUTA DI       Q      UN  GRAVE"
210 GOSUB 3030: PRINT "♥Q"
220 PRINT TAB(7)"T" TAB(12)"Y": PRINT "Q"
230 FOR T=0 TO 14
240 Y=5.2*T*T: PRINT TAB(6)T TAB(11)INT(Y)
250 NEXT T
260 GOSUB 3040: GRAPHIC 3: COLOR 6,4,1,0
270 FOR T=0 TO 14
280 Y=5.2*T*T: POINT 1, 512,Y
290 NEXT T
300 GOSUB 3040: GRAPHIC 0: COLOR 1,5,4,0
310 PRINT "♥QQQQQQQQQQQ   AL  RALLENTATORE"
320 GOSUB 3030: GRAPHIC 3: COLOR 6,4,1,0
330 FOR T=0 TO 14
340 Y=5.2*T*T: POINT 1, 512,Y
350 FOR H=1 TO 400: NEXT H
360 NEXT T
370 GOSUB 3040: GRAPHIC 0: COLOR 1,5,4,0
380 PRINT "♥QQQQQQQQQQQ   MOTO  PARABOLICO": GOSUB 3030
390 PRINT: PRINT TAB(4)"T" TAB(10)"X" TAB(16)"Y": PRINT
400 FOR T=0 TO 14
410 X=150+40*T: Y=5.2*T*T
420 PRINT TAB(3)T TAB(8)X TAB(15)INT(Y)
430 NEXT T: PRINT
440 GOSUB 3040: GRAPHIC 3: COLOR 6,4,1,0
450 FOR T=1 TO 14
460 X=150+40*T: Y=5.2*T*T: POINT 1, X,Y
470 FOR V=1 TO 200: NEXT V
480 NEXT T: GOSUB 3040
490 REM --------------------
500 REM   ASTA ORIZZONTALE
510 REM --------------------
520 GRAPHIC 0: COLOR 1,5,4,0
530 PRINT "♥QQQQQQQQQQQ       ASTA         Q     ORIZZONTALE"
540 GOSUB 3030: GRAPHIC 3: COLOR 6,4,1,0
550 FOR T=1 TO 14
560 XG=150+40*T: YG=5.2*T*T: X1=XG-110: X2=XG+110
570 DRAW 1, X1,YG TO X2,YG
580 FOR F=1 TO 200: NEXT F
590 NEXT T: GOSUB 3030
600 REM ------------------
610 REM   ASTA VERTICALE
620 REM ------------------
```

```
630 GRAPHIC 0: COLOR 1,5,4,0
640 PRINT "]QQQQQQQQQQ         ASTA          X      VERTICALE"
650 GOSUB 3030: GRAPHIC 3: COLOR 6,4,1,0
660 FOR T=0 TO 14
670 XG=150+50*T: YG=5.2*T*T+170
680 DRAW 1, XG,YG-160 TO XG,YG+200
690 FOR H=1 TO 200: NEXT H
700 NEXT T: GOSUB 3030
710 GRAPHIC 0: POKE 631,131: POKE 198,1: END
3000 REM =================
3010 REM   SOTTOPROGRAMMI
3020 REM =================
3030 FOR J=1 TO 500: GOSUB 3050: NEXT J: RETURN
3040 FOR J=1 TO 1000: GOSUB 3050: NEXT J: RETURN
3050 GET A$: IF A$="" THEN RETURN
3060 GET A$: IF A$="" THEN 3060
3070 RETURN

READY.
```

Listato 1 - Prima parte del programma.

```
720 REM ======================
730 REM   II FASE - ROTAZIONE
740 REM ======================
750 GRAPHIC 0: COLOR 1,5,4,0
760 PRINT "]QQQQQQQQQQ      II FASE       X      ROTAZIONE": GOSUB 3030
770 REM ----------------
780 REM   CIRCONFERENZA
790 REM ----------------
800 GRAPHIC 0: COLOR 1,5,4,0
810 PRINT "]QQQQQQQQQQ    CIRCONFERENZA": GOSUB 3030
820 GRAPHIC 3: COLOR 6,4,1,0
830 FOR K=0 TO 98 STEP 2
840 A=K*0.0628: X=512+220*COS(A): Y=512+332*SIN(A)
850 POINT 1, X,Y
860 NEXT K: GOSUB 3040
870 REM -----------------
880 REM   ASTA CHE RUOTA
890 REM -----------------
900 GRAPHIC 0: COLOR 1,5,4,0
910 PRINT "]QQQQQQQQQQ    ASTA CHE RUOTA": GOSUB 3030
920 GRAPHIC 3: COLOR 6,4,1,0
930 FOR K=0 TO 90 STEP 10
940 X=512+220*COS(0.0628*K): Y=512+332*SIN(0.0628*K)
950 DRAW 1, 512,512 TO X,Y
960 FOR J=1 TO 400: NEXT J
970 NEXT K: GOSUB 3040
980 REM  -----------------------------
990 REM    ASTA CHE RUOTA E SI SPOSTA
1000 REM -----------------------------
1010 GRAPHIC 0: COLOR 1,5,4,0
1020 PRINT "]QQQQQQQQQQ    ASTA CHE RUOTA     X     E SI SPOSTA": GOSUB 3030
1030 GRAPHIC 3: COLOR 6,4,1,0
1040 FOR K=0 TO 50 STEP 10
1050 X=220*SIN(0.0314*K): Y=332*COS(0.0314*K)
1060 X0=36+15*K: Y0=426: X1=X0+X: Y1=Y0-Y
1070 DRAW 1, X0,Y0 TO X1,Y1
1080 FOR J=1 TO 300: NEXT J
1090 NEXT K: GOSUB 3040
1100 REM -------------------------
1110 REM  ASTA CHE CADE RUOTANDO
1120 REM -------------------------
1130 GRAPHIC 0: COLOR 1,5,4,0
1140 PRINT "]QQQQQQQQQQ    ASTA CHE CADE     X       RUOTANDO": GOSUB 3030
1150 GRAPHIC 3: COLOR 6,4,1,0
1160 FOR T=0 TO 9
1170 FOR H=1 TO 500: NEXT H
1180 XG=150+80*T: YG=7.95*T*T+160
1190 X=110*SIN(0.314*T): Y=160*COS(0.314*T)
1200 DRAW 1, XG+X,YG-Y TO XG-X,YG+Y
1210 NEXT T: GOSUB 3040
1220 REM --------------
1230 REM   CON MURETTO
1240 REM --------------
1250 GRAPHIC 0: COLOR 1,5,4,0
1260 PRINT "]QQQQQQQQQQ      CON  MURETTO": GOSUB 3030
1270 GRAPHIC 3: COLOR 1,3,4,0: GOSUB 5030
1280 FOR T=0 TO 9
1290 FOR H=1 TO 500: NEXT H
1300 XG=150+80*T: YG=7.95*T*T+160
1310 X=110*SIN(0.314*T): Y=160*COS(0.314*T)
1320 DRAW 1, XG+X,YG-Y TO XG-X,YG+Y
1330 NEXT T: GOSUB 3040
1340 GRAPHIC 0: POKE 631,131: POKE 198,1: END
3000 REM =================
3010 REM   SOTTOPROGRAMMI
3020 REM =================
3030 FOR J=1 TO 500: GOSUB 3050: NEXT J: RETURN
3040 FOR J=1 TO 1000: GOSUB 3050: NEXT J: RETURN
3050 GET A$: IF A$="" THEN RETURN
3060 GET A$: IF A$="" THEN 3060
3070 RETURN
5000 REM --------------
5010 REM     MURETTO
5020 REM --------------
5030 REGION 2
5040 FOR H=323 TO 1023 STEP 6: DRAW 1, 0,H TO 150,H: NEXT H
5050 FOR K=363 TO 1023 STEP 40: DRAW 0, 0,K TO 150,K: NEXT K
5060 FOR K=363 TO 983 STEP 80: DRAW 0, 50,K TO 50,K+40
5070 DRAW 0, 100,K TO 100,K+40: NEXT K
5080 FOR K=323 TO 983 STEP 80: DRAW 0, 25,K TO 25,K+40
5090 DRAW 0, 75,K TO 75,K+40: DRAW 0, 125,K TO 125,K+40: NEXT K: REGION 5
5100 FOR K=950 TO 1023 STEP 6: DRAW 1, 160,K TO 1023,K: NEXT K: REGION 6
5110 RETURN

READY.
```

Listato 2 - Seconda parte del programma.

```
1350 REM ========================
1360 REM   III FASE - ANIMAZIONE
1370 REM ========================
1380 GRAPHIC 0: COLOR 1,5,4,0
1390 PRINT "]QQQQQQQQQQ      III FASE       X     ANIMAZIONE": GOSUB 3030
1400 REM --------------------
1410 REM   PUNTO CHE SI MUOVE
1420 REM --------------------
1430 PRINT "]QQQQQQQQQQ      PUNTO CHE      X      SI  MUOVE": GOSUB 3030
1440 GRAPHIC 3: COLOR 6,4,1,0
1450 FOR K=5 TO 1020 STEP 5
1460 POINT 0, K-5,512: POINT 1, K,512
1470 FOR L=1 TO 30: NEXT L
1480 NEXT K: GOSUB 3030
1490 REM -------------------
1500 REM   ASTA CHE SI MUOVE
1510 REM -------------------
1520 GRAPHIC 0: COLOR 1,5,4,0
1530 PRINT "]QQQQQQQQQQ      ASTA CHE       X       SI MUOVE": GOSUB 3030
1540 GRAPHIC 3: COLOR 6,4,1,0
1550 FOR K=0 TO 980 STEP 20
1560 XN=40+K: DRAW 0, X,350 TO X,700
1570 X=XN: DRAW 1, X,350 TO X,700
1580 FOR L=1 TO 150: NEXT L
1590 NEXT K: GOSUB 3030
1600 REM ----------------
1610 REM   ASTA CHE RUOTA
1620 REM ----------------
1630 GRAPHIC 0: COLOR 1,5,4,0
1640 PRINT "]QQQQQQQQQQ        ASTA        X       CHE RUOTA": GOSUB 3030
1650 GRAPHIC 3: COLOR 6,4,1,0: X=0: Y=0
1660 FOR K=0 TO 100 STEP 4
1670 XN=512+220*COS(0.0628*K): YN=512+332*SIN(0.0628*K): GOSUB 4040
1680 X=XN: Y=YN: GOSUB 4050
1690 FOR I=1 TO 200: NEXT I
1700 NEXT K: GOSUB 3030
1710 REM ---------------
1720 REM   ASTA CHE CADE
1730 REM ---------------
1740 GRAPHIC 0: COLOR 1,5,4,0: PRINT "]QQQQQQQQQQ";
1750 PRINT "        ASTA        X      CHE CADE       X      E RISALE"
1760 GOSUB 3030: GRAPHIC 3: COLOR 1,3,6,0: GOSUB 5030
1770 X1=140: Y1=160: X=0: Y=160
1780 FOR T=0 TO 9: M=1: GOSUB 4000
1790 FOR L=1 TO 150: NEXT L: GOSUB 4020
1800 X1=XG: Y1=YG: X=XN: Y=YN: GOSUB 4030
1810 NEXT T: GOSUB 6030: GOSUB 3030
1820 REM ------------------
1830 REM   ASTA CHE RISALE
1840 REM ------------------
1850 FOR T=9 TO 0 STEP -1: GOSUB 4000: GOSUB 4020
1860 X1=XG: Y1=YG: X=XN: Y=YN: GOSUB 4030
1870 FOR L=1 TO 150: NEXT L
1880 NEXT T: GOSUB 6030: GOSUB 3040
1890 REM --------------
1900 REM     TRACCIA
1910 REM --------------
1920 GRAPHIC 0: COLOR 1,5,4,0
1930 PRINT "]QQQQQQQQQQ        TRACCIA": GOSUB 3030
1940 GRAPHIC 3: COLOR 1,3,6,0: GOSUB 5030
1950 FOR T=0 TO 9 STEP 0.25: M=1: GOSUB 4000: GOSUB 4020
1960 POINT 1, X1+X,Y1-Y
1970 X1=XG: Y1=YG: X=XN: Y=YN: GOSUB 4030
1980 FOR L=1 TO 50: NEXT L
1990 NEXT T: GOSUB 6030: GOSUB 3040: GRAPHIC 0: END
3000 REM =================
3010 REM   SOTTOPROGRAMMI
3020 REM =================
3030 FOR J=1 TO 500: GOSUB 3050: NEXT J:RETURN
3040 FOR J=1 TO 1000: GOSUB 3050: NEXT J: RETURN
3050 GET A$: IF A$="" THEN RETURN
3060 GET A$: IF A$="" THEN 3060
3070 RETURN
4000 XG=150+80*T: YG=7.95*T*T+160
4010 XN=110*SIN(M*T): YN=160*COS(M*T): RETURN
4020 DRAW 0, X1+X,Y1-Y TO X1-X,Y1+Y: RETURN
4030 DRAW 1, X1+X,Y1-Y TO X1-X,Y1+Y: RETURN
4040 DRAW 0, 512,512 TO X,Y: RETURN
4050 DRAW 1, 512,512 TO X,Y: RETURN
5000 REM --------------
5010 REM     MURETTO
5020 REM --------------
5030 REGION 2
5040 FOR H=323 TO 1023 STEP 6: DRAW 1, 0,H TO 150,H: NEXT H
5050 FOR K=363 TO 1023 STEP 40: DRAW 0, 0,K TO 150,K: NEXT K
5060 FOR K=363 TO 983 STEP 80: DRAW 0, 50,K TO 50,K+40
5070 DRAW 0, 100,K TO 100,K+40: NEXT K
5080 FOR K=323 TO 983 STEP 80: DRAW 0, 25,K TO 25,K+40
5090 DRAW 0, 75,K TO 75,K+40: DRAW 0, 125,K TO 125,K+40: NEXT K: REGION 5
5100 FOR K=950 TO 1023 STEP 6: DRAW 1, 160,K TO 1023,K: NEXT K: REGION 6
5110 RETURN
6000 REM -------------------
6010 REM   SEGNALE ACUSTICO
6020 REM -------------------
6030 SOUND 0,0,240,0,15: FOR L=1 TO 50: NEXT L
6040 SOUND 0,0,0,0,0: RETURN

READY.
```

Listato 3 - Terza parte del programma.

Asta che cade per VIC 20 con super expander

60	La prima istruzione controlla quattro registri colore. Vengono assegnati allo sfondo, al bordo e al testo rispettivamente i colori bianco, verde e porpora; il quarto registro, che contiene zero, è quello del coiore ausiliario e in questo programma non viene usato. La seconda istruzione cancella lo schermo, abbassa il cursore di undici righe e scrive il titolo. La terza, porta il programma alla subroutine 3030 che crea una temporizzazione.
150	Il comando GRAPHIC 3 attiva la pagina grafica ad alta risoluzione. Lo schermo può essere pensato come una griglia di 1024 x 1024 posizioni, con l'origine in alto a sinistra, ed è a questa griglia che dobbiamo fare riferimento se vogliamo disegnare sullo schermo. Questo però non vuol dire che il VIC ha 1024 x 1024 punti. La sua massima risoluzione, ottenuta con la scheda, è di 160 x 160 punti, poichè coordinate che differiscono di 6 o 7 unità (a seconda della posizione sullo schermo) fanno disegnare lo stesso punto. Vi è ancora una considerazione da fare. Il numero dei punti in X e in Y è uguale, ma lo schermo è rettangolare con rapporto tra i lati di circa 3/2. Ciò significa che se vogliamo disegnare un quadro o un cerchio, dovremo moltiplicare la X per 0.66 oppure la Y per 1.5 (vedi Z nella linea 130).
160	POINT 1 serve a disegnare uno o più punti in una determinata posizione dello schermo. La sintassi è: POINT 1, X, Y dove X e Y sono le coordinate del punto. Qui disegna il punto di coordinate X = 150 e Y = 339. L'istruzione CHAR viene adoperata per scrivere nella pagina grafica. La sintassi è: CHAR R, C, "T" dove R è il numero di riga (da 0 a 19), c'è il numero della colonna (da 0 a 19) e T è il testo che può essere formato da lettere, numeri o caratteri grafici. In questo caso scrive alla riga 8 e colonna 0 le coordinate del punto appena disegnato.
180	DRAW 1 disegna un segmento di retta tra due punti di coordinate assegnate. La sintassi è: DRAW 1, X, Y TO X1, Y1 dove X, Y e X1, Y1 sono le coordinate degli estremi del segmento.
190	L'istruzione GOSUB 3040 crea un ciclo di temporizzazione di durata doppia del precedente. Il comando GRAPHIC 0 riporta il computer in testo.
260	Viene riattivata la pagina grafica ad alta risoluzione e vengono cambiati i colori assegnando il blu allo sfondo, il porpora al bordo e il bianco al testo.
710	Riporta il computer in testo. Carica dall'unità nastro automaticamente la seconda parte del programma, che deve essere stata memorizzata di seguito alla prima, e la manda in esecuzione.
840	Equazioni parametriche della circonferenza di centro 512, e di raggio 220, che diventa 332 in Y a causa del formato rettangolare dello schermo.
1460	POINT 0, X, Y cancella il punto di coordinate X, Y 0, per essere più precisi, lo disegna nello stesso colore dello sfondo, il che equivale a una cancellazione.
1560	L'istruzione DRAW0 è analoga alla precedente.
1810	GOSUB 6030 produce un segnale acustico.
3030-3040	Cicli di temporizzazione di durata diversa, con possibilità di arresto del ciclo durante la fase stessa di temporizzazione, premendo un tasto qualunque. Il ciclo prosegue ripremendo un tasto. All'inizio alla variabile J viene assegnato il valore 1; il programma passa poi alla subroutine 3050; se non viene premuto nessun tasto si ha il RETURN. In caso contrario il programma salta alla linea 3060 e vi rimane sino a quando non viene ripremuto un tasto. Passa poi al J successivo per ultimare il ciclo FOR-NEXT e quindi al RETURN.
5030	Viene cambiato solo il colore del testo; disegna ora in colore rosso.
6030	L'istruzione controlla cinque registri sonori. I primi tre sono oscillatori di nota e differiscono l'uno dall'altro di una ottava; il quarto è il generatore di rumore e il quinto controlla il volume. Ognuno di questi oscillatori può generare 128 diverse frequenze e i valori contenuti in questi registri sono quelli usati nei normali registri del suono del VIC. In questo particolare caso viene attivato uno dei tre oscillatori, al massimo volume, per il tempo prefissato dal ciclo FOR-NEXT.

Figura 1 - Le parti principali in cui è diviso il programma.

VIC-20

Spacca mattoni

Un divertente e famosissimo gioco realizzato per VIC 20

a cura della **Redazione**

Il gioco propone una versione per il VIC 20, nella configurazione minima di memoria, dell'ormai famoso *Breakout*: guidare una pallina che rimbalza verso la distruzione completa di un muro. Il movimento della paletta è stato codificato in linguaggio macchina per ovvie esigenze di velocità. Per capirlo in dettaglio, potete guardare la figura 1 dove sono riportate le istruzioni assembler; queste vengono inserite, in codifica decimale, nelle locazioni che vanno dalla 680 alla 755 (area disponibile per piccole routine).

```
      LDY (674)                  INY
      LDA #32                B   LDA #224
      STA (8141),Y               STA (8141),Y
      STA (8142),Y               STA (8142),Y
      STA (8143),Y               STA (8143),Y
      LDX #0                     STY (674)
      LDA (0),X                  RTS
      CMP (675)                  LDX #0
      BNE, A                 C   LDA (7680),X
      CPY #2                     CMP #204
      BEQ, A                     BEQ, D
      DEY                        INX
      LDA (0),X                  CPX #255
  A   CMP (676)                  BNE, C
      BNE, B                     LDA #1
      CPY #19                    STA (673)
      BEQ, B                 D   RTS
```

Figura 1 - Il simbolico delle due routine Assembler.

10-60	*Il codice macchina viene depositato mediante un normale ciclo che include READ ... DATA ... POKE.*
70	*Nella locazione 674 è memorizzata l'ultima posizione della paletta, mentre nelle locazioni libere in pagina zero (0 e 1) vi è l'indirizzo dove prelevare il codice del tasto premuto; il suo valore è memorizzato nel byte 675 se si tratta della Z, nel byte 676 per la C.*
80	*Da S1 a S4 sono tutti registri per il controllo dello schermo; T3 è l'indirizzo del generatore di tonalità più bassa; V0 l'indirizzo del registro di volume.*
90	*Pulito lo schermo si settano i colori per il bordo e lo sfondo.*
110-160	*L'intestazione e le istruzioni di gioco.*
170-175	*È in pratica il livello di difficoltà: più è alto il numero di palette a disposizione e più è semplice distruggere tutto il muro. Una risposta non corretta vi dà automaticamente una sola paletta.*
180	*Y contiene la posizione verticale della pallina, DY la sua direzione (− 1 equivale a "in alto", 1 a "in basso").*
190	*Viene azzerato il punteggio (PU).*
200-230	*Viene creato in rosso tutto il bordo del campo.*
240-260	*Stampa delle cinque file di mattoni: ciascuna ha un colore ed un punteggio diverso. D è il numero di palette che avete ancora in credito, oltre ovviamente a quella con cui state giocando.*
270	*Situazione punteggio, punteggio massimo e palette ancora disponibili.*
280	*È la routine in linguaggio macchina che preleva la vostra eventuale mossa e sposta la paletta. La velocità di quest'ultima dipende dal numero di istruzioni SYS 680 che si trovano all'interno del programma compreso tra la linea 260 e 360.*
290	*Cancella la vecchia posizione della pallina.*
300-340	*Viene calcolata la sua nuova posizione, controllando se vi è un rimbalzo laterale (DX = −DX), se la pallina ha urtato il fondo o qualche mattone.*
355	*Con SYS 736 parte la seconda routine Assembler che conta quanti mattoni restano ancora da demolire; se non ve ne sono più, viene messo un 1 nella locazione 673, che altrimenti contiene 0.*
360	*Altra mossa del giocatore, quindi si ritorna alla stampa della pallina, al punteggio e così via ...*
400-420	*La pallina ha urtato il fondo. Generazione del suono di commento. Viene decrementato il numero di palline che restano: se queste sono un numero maggiore o uguale a zero, potete ancora continuare (invertendo i valori di direzione X e DX, Y e DY), altrimenti si salta alla linea 700.*
450-500	*Sono stati distrutti tutti i mattoni e ciò vi viene sottolineato visivamente e musicalmente. Viene anche generato un bonus che tiene conto del punteggio e del livello di difficoltà.*
510-720	*Scritte di conclusione e la richiesta per una nuova partita.*
3000-3020	*Questa subroutine genera un valore casuale per X e DX; se DX = −1 la pallina parte a sinistra, altrimenti (DX = 1) va a destra.*

Figura 2 - Note relative al programma.

Figura 3 - Il listato per il VIC 20.

```
*** SPACCAMATTONI ***

 10 FORI=1TO55:READA:POKE679+I,A:NEXT
 20 FORI=1TO20:READA:POKE735+I,A:NEXT
 30 DATA172,162,2,169,32,153,205,31,153
,206,31,153,207,31,162,0,161,0,205,163,
2,208,7
 40 DATA192,2,240,3,136,161,0,205,164,2
,208,5,192,19,240,1,200,169,224,153,205
,31,153
 50 DATA206,31,153,207,31,140,162,2,96
 60 DATA162,0,189,0,30,201,204,240,10,2
32,224,255,208,244,169,1,141,161,2,96
 70 POKE0,197:POKE1,0:POKE673,0:POKE674
,10:POKE675,33:POKE676,34
 80 A$="              ":S1=36864:S2=1+S1
:S3=S2+1:S4=S3+1:T3=36876:VO=T3+2:T2=T3
-1
```

```
 90 PRINT"":POKEVO+1,253
 110 PRINT"        SPACCAMATTONI"
 120 PRINT"  DISTRUGGI TUTTI  I MATT
ONI DEL MURO, OGNI FILA DA' UN PUNTEGGI
O DIVERSO"
 130 PRINT" CON Z VAI A SINISTRA"
 140 PRINT" CON C VAI A DESTRA"
 150 PRINT"   "CHR$(28)" PREMI UN TA
STO "
 160 GETB$:IFB$=""THEN160
 170 PRINT""CHR$(28)" QUANTE PALETT
E VUOI":INPUT"    (1,2,3,4)";L
 175 IFL<1ORL>4THENL=1
 180 POKEVO+1,239:POKEVO,15:Y=20:DY=-1:
GOSUB3000:PRINT"":IFPU>PMTHENPM=PU
 190 PU=0
 200 FORI=7680+44TO7680+44+21:POKEI,224
 205 POKEI+30720,2:NEXT
 210 FORI=3TO22:POKE7680+22*I,208:POKE3
```

```
8400+22*I,2:POKE7701+22*I,207
 220 POKE38421+I*22,2:NEXT
 230 FORI=0TO21:POKE8164+I,207:POKE3888
4+I,2:NEXT
 240 FORI=3TO7:FORJ=1TO20
 250 POKE7680+22*I+J,204:POKE38400+22*I
+J,I:NEXTJ,I:D=L-1
 260 POKE7680+X+22*Y,81
 270 PRINT"▒▒PUNTI■:"PU;TAB(12)"P.MAX:
"PM"▒▒▒▒▒▒▒CREDITI:"D"■"
 280 SYS680
 290 POKE7680+X+22*Y,32
 300 X=X+DX
 310 IFX<2ORX>19THENDX=-DX:POKET3,220
 320 Y=Y+DY:IFY=22THEN400
 330 IFPEEK(7658+X+22*Y)=204THENDY=1:PO
KET3,230:POKE7658+X+22*Y,32:PU=PU+(10-Y
)
 340 IFPEEK(7702+X+22*Y)=224ORPEEK(7658
+X+22*Y)=224THENDY=-DY:POKET3,240
 350 POKET3,0
 355 IFPU>190THENSYS736:IFPEEK(673)=1TH
EN450
 360 SYS680:GOTO260
 400 POKEVO,15:FORI=230TO160STEP-.15:PO
KET2-1,I:NEXT:POKET2-1,0
 410 D=D-1:IFD>-1THENY=20:DY=-1:GOSUB30
00:GOTO260
 420 GOTO700
 450 PRINT"▒":POKEVO+1,27:FORI=160TO240
STEP.5:POKET3,I:B%=RND(1)*505+1
 460 POKE7680+B%,160:POKE38400+B%,INT(R
ND(1)*7)+1
 470 NEXT:POKET3,0:Q%=(PU/(L*3)+.5):I=0
:J=0
 480 J=J+1:FORI=0TO22:POKES1,12+I:POKES
2,38+I:POKES3,150-I:POKES4,174-I*2:POKE
T3,220+I:NEXT
 490 FORI=22TO0STEP-1:POKES1,12+I:POKES
2,38+I:POKES3,150-I:POKES4,174-I*2:POKE
T3,220+I:NEXT
 500 IFJ<5THEN480
 510 POKET3,0:PRINT"▒▒▒▒VINCI UN BONUS
 DI "Q%" PUNTI IL PUNTEGGIO E' ORA "PU+
Q%:PU=PU+Q%
 520 FORI=0TO4000:NEXT
 530 PRINT"▒▒▒▒COMPLIMENTI■, CE L'HAI F
ATTA!!! IL TUO PUNTEGGIO E' STATO DI ▒
"PU"■."
 540 IFPU>PMTHENPRINT"▒▒▒BRAVO■■ PERCHE
' E' ANCHE IL MIGLIORE !"
 550 GOTO710
 700 PRINT"▒▒▒▒▒▒▒▒N. PALETTE "D+1:POKET
3,0:POKET2,0
 710 POKE198,0:PRINTA$"    ■ANCORA (S/N)
";
 720 INPUTR$:IFLEFT$(R$,1)="S"THEN180
 740 PRINT"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒"CHR$(28)"C
IAO■":END
 3000 X=INT(RND(1)*20)+1:IFX<2ORX>19THE
N3000
 3010 DX=INT(RND(1)*3)-1:IFDX=0THEN3010
 3020 RETURN
```

```
Lista simboli grafici

  80 : 1  HOME =CHR$(19)
      11  CRSR↑ =CHR$(17)
       1  CRSR← =CHR$(29)

  90 : 1  SHIFT HOME =CHR$(147)

 110 : 1  CRSR↑ =CHR$(17)
       1  REVERSE =CHR$(18)
       1  SHIFT REVERSE =CHR$(146)

 120 : 2  CRSR↑ =CHR$(17)

 130 : 2  CRSR↑ =CHR$(17)

 140 : 1  CRSR↑ =CHR$(17)

 150 : 2  CRSR↑ =CHR$(17)
       1  REVERSE =CHR$(18)
       1  SHIFT REVERSE =CHR$(146)

 170 : 1  SHIFT HOME =CHR$(147)
       4  CRSR↑ =CHR$(17)

 180 : 1  SHIFT HOME =CHR$(147)

 270 : 1  HOME =CHR$(19)
       1  REVERSE =CHR$(18)
       1  SHIFT REVERSE =CHR$(146)
       1  HOME =CHR$(19)
       1  CRSR↑ =CHR$(17)
       5  CRSR← =CHR$(29)
       1  REVERSE =CHR$(18)
       1  SHIFT REVERSE =CHR$(146)

 450 : 1  SHIFT HOME =CHR$(147)

 510 : 4  CRSR↑ =CHR$(17)

 530 : 1  SHIFT HOME =CHR$(147)
       1  CRSR↑ =CHR$(17)
       1  REVERSE =CHR$(18)
       1  SHIFT REVERSE =CHR$(146)
       1  REVERSE =CHR$(18)
       1  SHIFT REVERSE =CHR$(146)

 540 : 1  CRSR↑ =CHR$(17)
       1  REVERSE =CHR$(18)
       1  SHIFT REVERSE =CHR$(146)

 700 : 1  HOME =CHR$(19)
       1  CRSR↑ =CHR$(17)
       5  CRSR← =CHR$(29)

 740 : 1  SHIFT HOME =CHR$(147)
       6  CRSR↑ =CHR$(17)
       9  CRSR← =CHR$(29)
       1  REVERSE =CHR$(18)
       1  SHIFT REVERSE =CHR$(146)
```

Il gioco della scimmia

riflessi pronti, attenzione, memoria: ecco le qualità di una brava ... scimmia

a cura della **Redazione**

È anche questo un gioco molto noto, seppure ancora giovane. Si tratta di "scimmiottare" il calcolatore esercitando la nostra memoria visiva e sonora.
Il programma è stato scritto per il VIC 20 (vedi figura 1), ma può benissimo essere modificato per un altro computer.
Ma vediamo qual'è il principio di funzionamento: si accendono uno per volta, in quattro diverse posizioni dello schermo, altrettanti quadrati, ciascuno di un colore diverso. Sono stati usati il porpora, il verde, il blu e il giallo, cui corrispondono i tasti 5, 6, 7, 8 (senza schiacciare il tasto *CTRL*); il colore dei singoli flash, e quindi il suo colore, è generato casualmente. In pratica bisogna solo ripetere la sequenza premendo i corrispondenti tasti coi colori; ma vediamo in dettaglio il funzionamento di ciascun gioco.

Gioco 1: create la sequenza

Dopo aver ripetuto la prima nota che il VIC vi fornisce, siete voi a dover aggiungere una nuova nota. Continuando in questo modo dovete ripetere le prime due note e quindi aggiungerne ancora una, e così via finché non avrete esaurito il numero di passi stabiliti per ciascun livello di difficoltà (vedi tabella 1).
Il VIC sottolineerà la vittoria con le prime otto note della Quinta Sinfonia di Beethoven. Se invece non siete stati abbastanza veloci o avete commesso un errore, il gioco finisce lasciandovi però la possibilità di sentire l'ultima sequenza corretta.
Prima di partire per una nuova partita avete sempre modo di modificare il livello di difficoltà, premendo i tasti funzione (vedi tabella 1).

Gioco 2: ripetete la sequenza

Il VIC genera la prima nota; ripetetela correttamente e il VIC ve ne aggiungerà una seconda, sempre secondo una generazione casuale, e così via finché non sarà stato raggiunto il numero di note corrispondente alla sequenza scelta.

Gioco 3: la sequenza che cambia

È molto simile al gioco 2. L'unica differenza è che il VIC non ripete la sequenza generata precedentemente, ma inizia sempre con una nuova, cui di volta in volta viene aggiunta una nuova nota.

Gioco 4: singoli flash

Il VIC genera una sola nota che bisogna ripetere entro 1 secondo.
Nella figura 2 trovate le sezioni più importanti del programma, in figura 3 invece la lista con le principali variabili usate.
I valori delle variabili *D1*, *N1* e *D2* sono fissati dalla pressione dei tasti funzione (vedi linee *120-150*). Potete ovviamente cambiarli secondo i vostri gusti e la vostra velocità.

Tasto funzione	Numero di note per la sequenza	Intervallo di tempo tra 2 note	Tempo entro cui bisogna rispondere
F1	4	100	5 sec.
F3	8	50	3 sec.
F5	16	10	2 sec.
F7	32	1	1 sec.

Tabella 1 - I livelli di difficoltà per ogni tipo di gioco.

40-200	*Inizializzazione. Scelta del tipo di gioco e del livello di difficoltà.*
220-280	*Le istruzioni relative al primo gioco.*
300-350	*Quelle relative al secondo.*
370-390	*Quelle relative al terzo.*
410-430	*Quelle relative al quarto.*
450-520	*La subroutine che genera il suono e il quadretto colorato per ogni flash.*
540-610	*Subroutine per la ripetizione della sequenza.*
630-660	*Le scritte finali nel caso di errore.*
670-700	*La musica di accompagnamento.*
720-760	*I commenti in caso di vittoria; il punteggio parziale e totale; le prime note della Quinta Sinfonia.*
780-860	*La possibilità di rivedere e risentire l'ultima sequenza.*
880-940	*La proposta per una nuova partita.*
960-980	*Per svuotare lo STACK.*

Figura 2 - Le parti principali in cui è diviso il programma.

BK	*La durata delle note.*
B$	*11 discese del cursore + il codice del porpora.*
C$	*I quattro colori corrispondenti alle posizioni di ciascun quadrato.*
D	*Indirizzo del generatore musicale di tonalità alta.*
D1	*Intervallo di tempo tra due flash.*
D2	*Tempo, in secondi, che avete a disposizione per rispondere con la nota giusta.*
D3	*Vi viene memorizzato il tempo di risposta, per confrontarlo poi con D2.*
G	*Il numero di partite con lo stesso tipo di gioco.*
K(N)	*Le posizioni dei quadrati colorati nell'N-esimo flash della sequenza.*
N1	*Il numero massimo di flash per quella sequenza.*
P	*Numero di flash nel gioco 4.*
Q	*Flag prima dell'uscita da una subroutine.*
V0	*Indirizzo del registro che controlla il volume.*
X	*Numero del gioco.*

Figura 3 - Le principali variabili usate.

Figura 1
Il listato per il VIC 20

```
GIOCO DELLA SCIMMIA

 40 POKE36879,13:PRINT""
 50 PRINT"QUATTRO TIPI DI GIOCO:
1.UNA TUA SEQUENZA    2.RIPETENDO LA"
 60 PRINT"SEQUENZA":PRINT" 3.CAMBIA
NDO SEQUENZA  4.SINGOLI FLASH"
 70 PRINT"     CHE NUMERO ? ";
 80 GETA$:IFA$=""THEN80
 90 X=VAL(A$):PRINTX
 100 PRINT"  PER INCREMENTARE IL  LIV
ELLO DI DIFFICOLTA'PREMI F1,F3,F5,F7
"
 110 GETA$:IFA$=""THEN110
 120 IFA$=CHR$(133)THEND1=100:N1=4:D2=5
 130 IFA$=CHR$(134)THEND1=50:N1=8:D2=3
 140 IFA$=CHR$(135)THEND1=10:N1=16:D2=2
 150 IFA$=CHR$(136)THEND1=1:N1=32:D2=1
 160 B$="":C$="PORP. VERDE
BLU GIALLO"
 170 DIMK(32):VO=36878:D=36876:S=7790:C
O=38510:G=0:DD=0
 180 NM=1:G=G+1:Q=0
 190 PRINT""B$C$:W=RND(-TI)
 200 ONXGOTO220,300,410
 220 PRINT"IL VIC SCEGLIE IL PRIMO COL
ORE":N=1:K(1)=INT(RND(1)*4):GOSUB450
 230 GOSUB540:IFQ<>0THEN960
 240 NM=NM+1:N=NM:PRINT"AGGIUNGI UN N
UOVO         COLORE":D3=TI
 250 GETA$:IFA$=""ANDTI-D3<D2*60THEN250
 260 IFTI-D3>D2*50THEN630
 270 K(N)=VAL(A$)-5:GOSUB450:GOSUB540:I
FQ<>0THEN960
 280 GOTO240
 300 N=NM:K(N)=INT(RND(1)*4):GOSUB450
 310 GOSUB540:IFQ<>0THEN960
 320 FORI=1TO800:NEXT
 330 FORN=1TONM:PRINT"IL VIC TI DA' Q
UESTA SEQUENZA":GOSUB450:NEXTN
 340 NM=NM+1:GOTO300
 350 FORI=1TO800:NEXT
 370 FORN=1TONM:PRINT"IL VIC TI DA' Q
UESTA SEQUENZA":K(N)=INT(RND(1)*4)
 380 GOSUB450:NEXTN:GOSUB540:IFQ<>0THEN
960
 390 FORI=1TO800:NEXT:NM=NM+1:GOTO370
 410 P=1
 420 D2=1:N=1:K(N)=INT(RND(1)*4):GOSUB4
50:GOSUB540:IFQ<>0THEN960
 430 FORI=1TO400:NEXT:NM=1:N=1:P=P+1:GO
TO420
 450 POKEVO,15:POKED,135+K(N)*25:PRINT"
"B$+"PASSO NUM.:";
 460 IFX=4THENPRINTP:GOTO480
 470 PRINTN
 480 FORJ=0TO4:FORL=CO+J*22+6*K(N)TOL+3
:POKEL,K(N)+4:NEXTL,J
 490 FORJ=0TO4:FORI=S+J*22+6*K(N)TOI+3:
POKEI,160:NEXTI,J
 500 FORJ=0TOD1:NEXT:POKED,0
 510 PRINT""B$C$
 520 RETURN
 540 FORN=1TONM:PRINT"  RIPETI LA SEQU
ENZA":D3=TI
 550 GETA$:IFA$=""ANDTI-D3<D2*60THEN550
 560 IFTI-D3>D2*55THENQ=1:RETURN
 570 Y=VAL(A$):IFY-5<>K(N)THENQ=2:RETUR
N
 580 IFY-5=K(N)THENGOSUB450:NEXTN
 590 IFX=4THENN=P+1
 600 IFN=N1+1THENQ=3:RETURN
 610 RETURN
 630 PRINT"HAI PERSO! NON SEI STATO
ABBASTANZA VELOCE":GOTO650
 640 PRINT"-ERRORE- HAI PERSO!"
 650 FORN=1TO1:NEXT:IFX=4THENNM=P
 660 PRINT"  "NM-1"    PASSI ERANO
 CORRETTI":PRINT" N. DI GIOCHI:"G
 670 POKEVO,15:RESTORE
 680 READW:IFW=-1THENPOKET+1,0:GOTO780
 690 READBK:POKET+1,W:FORI=1TOBK:NEXTI:
POKET+1,0:FORI=1TO10:NEXTI:GOTO680
 700 DATA175,100,175,100,175,100,151,50
0,0,300,163,100,163,100,163,100,147,500
,-1
 720 PRINT"COMPLIMENTI HAI FINITO
 CON"N1" PUNTI"
 730 PRINT" N. DI GIOCHI:"G:DD=DD+N1:P
RINT"IN TOTALE HAI"DD;
 740 PRINT"PUNTI":POKEVO,15:RESTORE
 750 READW:IFW=-1THEN780
 760 READBK:POKED,W:FORI=1TOBK:NEXTI:PO
KED,0:FORI=1TO10:NEXTI:GOTO750
 780 IFX=3ORX=4THEN880
 800 PRINT"VUOI VEDERE L'ULTIMA
SEQUENZA(S/N)"
 810 GETA$:IFA$=""THEN810
 820 IFA$<>"S"ANDA$<>"N"THEN810
 830 IFA$="S"THENPRINT""B$C$
 840 IFA$="N"THEN880
 850 IFN=N1+1THENNM=NM+1
 860 FORN=1TONM-1:GOSUB450:NEXTN
 880 PRINT"  VUOI CONTINUARE"
 885 PRINT"   1.ALLO STESSO MODO2
.CAMBIANDO LE"
 890 PRINT"CONDIZIONI":PRINT"3
. FINIRE":PRINT"   PREMI IL NUMERO"
 895 PRINT"DESIDERATO"
 900 GET A$:IFA$=""THEN900
 910 Y=VAL(A$):IFY<>1ANDY<>2ANDY<>3THEN
900
 920 ONYGOTO180,930,940
 930 RUN
 940 END
 950 REM -PULIZIA DELLO STACK-
 960 IFQ=1THENQ=0:GOTO630
 970 IFQ=2THENQ=0:GOTO640
 980 IFQ=3THENQ=0:GOTO720
```

Lista simboli grafici

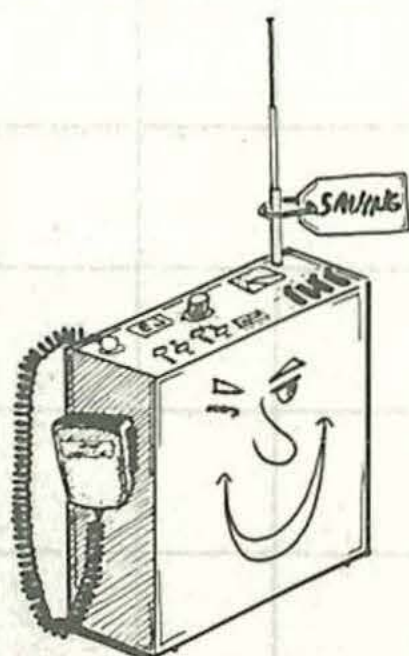
```
 50 : 1  SHIFT HOME =CHR$(147)
      2  CRSR↑ =CHR$(17)
      1  REVERSE =CHR$(18)
      1  SHIFT REVERSE =CHR$(146)
      2  CRSR↑ =CHR$(17)

 60 : 3  CRSR← =CHR$(29)
      2  CRSR↑ =CHR$(17)

 70 : 2  CRSR↑ =CHR$(17)
      1  REVERSE =CHR$(18)
      1  SHIFT REVERSE =CHR$(146)

100 : 2  CRSR↑ =CHR$(17)
      1  REVERSE =CHR$(18)
      3  CRSR← =CHR$(29)

160 : 10  CRSR↑ =CHR$(17)

190 : 1  SHIFT HOME =CHR$(147)

220 : 1  CRSR↑ =CHR$(17)

240 : 1  CRSR↑ =CHR$(17)
      1  CRSR← =CHR$(29)
```

Il gioco della scimmia

Seguito listato Il gioco della scimmia

```
330 : 1   CRSR↑ =CHR$(17)
      1   CRSR← =CHR$(29)

370 : 1   CRSR↑ =CHR$(17)
      1   CRSR← =CHR$(29)

450 : 1   HOME =CHR$(19)
      5   CRSR↑ =CHR$(17)
      3   CRSR← =CHR$(29)

510 : 1   SHIFT HOME =CHR$(147)

540 : 1   CRSR↑ =CHR$(17)

630 : 1   SHIFT HOME =CHR$(147)
      2   CRSR↑ =CHR$(17)

640 : 1   SHIFT HOME =CHR$(147)
      2   CRSR↑ =CHR$(17)

660 : 2   CRSR↑ =CHR$(17)

720 : 1   SHIFT HOME =CHR$(147)
      2   CRSR↑ =CHR$(17)
      1   REVERSE =CHR$(18)
      1   SHIFT REVERSE =CHR$(146)

730 : 2   CRSR↑ =CHR$(17)

800 : 2   CRSR↑ =CHR$(17)
      1   CRSR← =CHR$(29)
      1   REVERSE =CHR$(18)
      5   CRSR← =CHR$(29)
      1   SHIFT REVERSE =CHR$(146)

830 : 1   SHIFT HOME =CHR$(147)

880 : 1   SHIFT HOME =CHR$(147)
      3   CRSR↑ =CHR$(17)

885 : 1   REVERSE =CHR$(18)
      4   CRSR← =CHR$(29)
      1   CRSR↑ =CHR$(17)

890 : 1   REVERSE =CHR$(18)
      4   CRSR← =CHR$(29)
      1   REVERSE =CHR$(18)
      1   CRSR↑ =CHR$(17)
      2   CRSR← =CHR$(29)
      3   CRSR↑ =CHR$(17)
      1   SHIFT REVERSE =CHR$(146)

895 : 5   CRSR← =CHR$(29)
```

Questo programma è disponibile, insieme al programma "spacca mattoni", su cassetta versione VIC 20 a L. 15.000.
Per ordinazioni vedere il coupon "Servizio Software" a pagina 130.

SAVING COMPUTER '83
APPLE
DYSAN
SORD
SIRIUS
DIGITEK
COMMODORE
La sorgente
per le necessità del tuo computer
Nella nostra sala mostra
potrai ammirare e provare prodotti come:
stampanti
floppy disk
programmi
biblioteca specializzata
Le migliori marche di Personal Computer
Disponiamo infatti pronta consegna
di APPLE II, APPLE II E, SIRIUS,
SORD M 23, AVT COMP 2, VIC 20,
VIC 64, ZX81, SPECTRUM, MPF II
Non perdere
questa occasione!!!
DISTRIBUTORI ESCLUSIVI DEL FAVOLOSO
"THE LAST ONE" PER IL VENETO
Vendita anche per corrispondenza,
telefona per le quotazioni, saremo lieti di accontentarti.
SAVING
SAVING
ELETTRONICA
VIA GRAMSCI 40 - MIRANO (VE) - TEL. (041) 432876

Filetti e gravità

Pallini impilati per realizzare filetti di cinquine.

di G. Di Bella

Il gioco la cui prima apparizione risale, ad una edizione di Giochi senza Frontiere di almeno tre anni fa, e tutto sommato classico ma, si spera, divertente nella sua semplicità.
Due giocatori, a turno e contrassegnati da colori diversi, devono cercare di "depositare" su una scacchiera 8 x 8 cinque pallini consecutivi in file orizzontali, verticali o diagonali. Direte: "ma ci abbiamo già giocato un sacco di volte (a scuola)!".
Non si tratta di disegnare, come al solito, i pallini su di un normale foglio quadrettato, ma, ecco la novità, di sganciarli impilando ogni nuovo pallino sul vecchio, nelle diverse colonne della scacchiera.
In altre parole, osservando la figura 1, è possibile depositare il pallino A nella posizione (6,1) solo se sono già presenti, i pallini B(7,1) e C(8,1).
Il calcolatore si limita quindi a riconoscere, mediante la scansione completa della scacchiera, se qualcuno ha realizzato il filetto.
Il gioco gira sul VIC 20 nella sua configurazione base (3,5 Kbyte di memoria RAM) e, anche se il programma centrale in sè occupa ben poco spazio, le stringhe per la creazione della scacchiera, il corposo gruppo di variabili, la piccola presentazione e qualche altro gadget lasciano liberi in esecuzione poco più di 150 byte.

5-28	*La "macchinosa" intestazione del programma si avvale anche della SUBROUTINE 2000; notare quanto occorre per generare due secondi di presentazione. Seguono le istruzioni di gioco.*
30-200	*L'input dei nomi e la stampa del campo di gioco. Il contatore dei pallini depositati viene posto a zero.*
210-220	*Parte per primo il giocatore col pallino porpora.*
230-305	*Si preleva e si smista la mossa del giocatore.*
310-330	*Si impila il pallino nella sua colonna.*
340-360	*Dalla posizione video del pallino or ora depositato si risale alle coordinate X, Y per aggiornare la situazione della scacchiera (matrice A); viene quindi incrementato il numero di pallini depositati e, se questi sono minori di nove si salta tutta la fase di scansione della scacchiera.*
400-665	*È il programma vero e proprio: la ricerca dei cinque pallini in fila, dapprima sulle righe, poi sulle colonne e infine su tutte le diagonali con almeno cinque caselle.*
680	*Se la scacchiera si è riempita completamente e nessuno ha realizzato il filetto è ovvio che la partita si chiude in parità.*
800-810	*Il pallino è stato depositato senza provocare nulla. È allora la volta di abilitare al gioco l'altro giocatore (PL = 1 o 2).*
900	*Viene generato un suono che indica una mossa non corretta.*
850	*Genera il tonfo della pallina sul fondo.*
910	*Pulisce il buffer riservato alla tastiera dagli ultimi codici dei caratteri premuti.*
920-1090	*Proclamazione visiva e sonora del vincitore con la richiesta per un'eventuale nuova partita.*

Figura 3 - Note relative al programma.

A(8,8)	*Matrice che contiene la situazione della scacchiera: se in A(S,K) c'è zero vuol dire che nella posizione (S,K) non c'è nessuno, altrimenti indica la presenza di un pallino: vi viene memorizzato il suo codice-colore (5=pallino porpora 4=pallino verde).*
C	*Numero di pallini consecutivi durante la scansione della scacchiera.*
CL	*Codice-colore del giocatore attualmente abilitato.*
D	*Contatore del numero di pallini depositati.*
G1	*Indirizzo del generatore di suono con la tonalità più bassa.*
P(C)	*Vettore contenente le posizioni video dei pallini durante la scansione delle righe, delle colonne e delle diagonali.*
P1$	*Nome del primo giocatore.*
P2$	*Nome del secondo giocatore.*
M	*Codice del carattere premuto da tastiera.*
V	*Indirizzo del registro che controlla il volume.*

Figura 2 - Lista delle principali variabili usate.

Figura 1
Il campo di gioco.

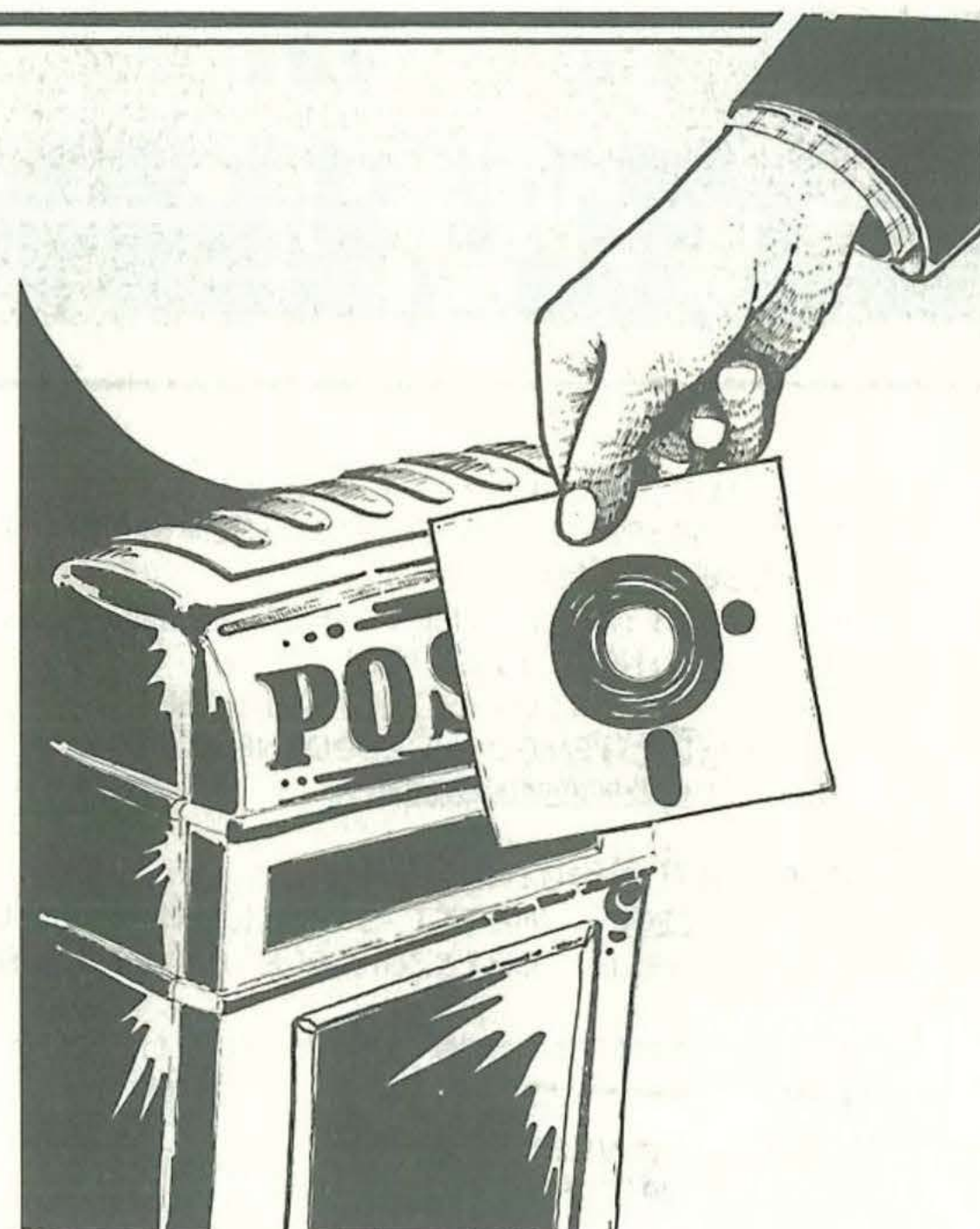
Copiate il listato con molta attenzione, senza omettere nulla, in particolare il *REMARK* della linea *5*, visto che i suoi caratteri vengono poi utilizzati nella creazione dell'intestazione.

Figura 4
Il listato BASIC.

```
 5 REM FILETTO GRAVITAZIONALE
 7 PRINT"[CLR]":P=7:C=5:A=4110:FL=30720:I=7
702:POKE36879,29
 10 POKEI,81:POKEI+FL,C:FORS=1TO14:M=2+
S:GOSUB2000:NEXT:I=7702
 15 C=4:P=5:A=4102:POKEI,81:POKEI+FL,C:
FORS=1TO7:M=5+S:GOSUB2000:NEXT
 20 PRINT"[DOWN x15]      DI G. DI BE
LLA"
 21 FORI=1TO3000:NEXT
 22 PRINT"[CLR][DOWN]     METTI 5 PALLINI IN FILA
: IN DIAGONALE, ORIZZONTALE O VERTICALE
"
 23 PRINT"[DOWN]CON [S] SI VA A SINISTRA[DOWN]C
ON [D] A DESTRA[DOWN]CON [F] SI 'SGA
NCIA'[DOWN]"
 27 PRINT"[DOWN]     PREMI UN TASTO"
 28 IFPEEK(197)=64THEN28
 30 CLR:GOSUB910:PRINT"[CLR][DOWN]":POKE3687
9,31
 40 INPUT"[DOWN] 1°NOME";P1$
 41 INPUT"[DOWN] 2°NOME";P2$
 50 A$=" ┌┬┬┬┬┬┬┬┐"
 55 B$="  ├┼┼┼┼┼┼┼┤"
 56 C$="  | | | | | | | | |"
 60 PRINT"[HOME][DOWN]"CHR$(28)"▒"A$"▒";
 70 FORI=0TO6:PRINTC$:PRINTB$:NEXT:PRIN
TC$
 80 PRINT"[DOWN]└┴┴┴┴┴┴┴┘"
 90 PRINT"[...]                                    "
 200 V=36878:G1=36874:SC=36879:D=0
 210 X=7746:PL=1:CL=4:S=44
 220 POKEX,81:POKEX+30720,CL
 225 FORW=0TO20:NEXT
 230 M=PEEK(197):IFM=64THEN230
 240 IFM=41THEN280
 250 IFM=42THEN300
 260 IFM<>18THEN230
 265 IFX=7767THENGOSUB900:GOTO230
 270 POKEX,32:X=X+1:GOTO220
 280 IFX=7746THENGOSUB900:GOTO230
 290 POKEX,32:X=X-1:GOTO220
 300 IFPEEK(X+S)<>32THENGOSUB900:GOTO22
0
 305 POKEX,32
 310 S=S+44:IFS>352THEN330
 320 IFPEEK(X+S)=32THENPOKEX+S-44,32:PO
KEX+S,81:POKEX+S+30720,CL:GOTO310
 330 POKEX+S-44,81:POKEX+S-44+30720,CL:
GOSUB850
 340 I=(S-44)/44
 350 J=(X-7749)/2+1
 360 A(I,J)=CL:D=D+1:IFD<9THEN800
 400 C=0:FORK=1TO8
 410 IFA(I,K)<>CLTHENC=0:GOTO430
 420 C=C+1:P(C)=7749+I*44+(K-1)*2:IFC=5
THEN1000
 430 NEXT
 450 IFI>4THEN500
 460 C=0:FORK=1TO8
 470 IFA(K,J)<>CLTHENC=0:GOTO495
 480 C=C+1:P(C)=7749+K*44+(J-1)*2:IFC=5
THEN1000
 495 NEXT
 500 FORH=5TO8:C=0:FORK=HTO1STEP-1:S=H-
K+1
 510 IFA(S,K)<>CLTHENC=0:GOTO530
 520 C=C+1:P(C)=7749+S*44+(K-1)*2:IFC=5
THENGOTO1000
 530 NEXTK,H
 550 FORH=2TO4:C=0:FORK=8TOHSTEP-1:S=H-
K+8
 560 IFA(S,K)<>CLTHENC=0:GOTO580
 570 C=C+1:P(C)=7749+S*44+(K-1)*2:IFC=5
THENGOTO1000
 580 NEXTK,H
 600 FORH=4TO1STEP-1:C=0:FORK=HTO8:S=K-
H+1
 610 IFA(S,K)<>CLTHENC=0:GOTO630
 620 C=C+1:P(C)=7749+S*44+(K-1)*2:IFC=5
THENGOTO1000
 630 NEXTK,H
 650 FORH=4TO2STEP-1:C=0:FORK=1TO9-H:S=
H+K-1
 655 IFA(S,K)<>CLTHENC=0:GOTO665
 660 C=C+1:P(C)=7749+S*44+(K-1)*2:IFC=5
THENGOTO1000
 665 NEXTK,H
 680 IFD=64THENPRINTCHR$(28)"[HOME]          P
ARITA'":GOSUB910:W=255:POKEV,15:GOTO104
6
 800 IFPL=1THENX=7767:CL=5:PL=2:S=44:GO
TO220
 810 GOTO210
 850 POKEV,15:POKEG1+2,170:FORW=0TO7:NE
XT:POKEG1+2,0:POKEV,0:RETURN
 900 POKEV,15:FORI=255TO128STEP-1:POKEG
1,I:NEXT:POKEG1,0:POKEV,0:RETURN
 910 FORI=0TO9:GETP1$:NEXT:RETURN
 920 POKEV,15:POKEG1+2,L+128:L=L+1:RETU
RN
 1000 PRINT"[HOME]BRAVO "CHR$(28);:IFPL=1T
HENPRINTP1$;:GOTO1020
 1010 PRINTP2$;
 1020 PRINT" HAI VINTO!":GOSUB910
 1025 FORJ=0TO8
 1030 FORI=1TO5:POKEP(I),32:GOSUB920:NE
XT
 1035 FORW=0TO150:NEXT
 1040 FORI=1TO5:POKEP(I),81:GOSUB920:NE
XT
 1045 FORW=0TO450:NEXTW,J:FORW=LTO254:P
OKEG1+2,W:NEXT
 1046 FORI=WTO0STEP-1:POKEG1+2,I:NEXT:P
RINT"[HOME]PER CONTINUARE PREMI C":POKEV,0:
POKEG1+2,0
 1050 GETP1$:IFP1$=""THEN1050
 1060 IFP1$="C"THEN30
 1070 POKE36881,159
 1080 PRINTCHR$(28)"[CLR][DOWN]GAME OVER
":POKEV,15
 1090 FORI=159TO39STEP-1:POKE36881,I:PO
KEG1+2,I+95:NEXT:POKEV,0:END
 2000 FORI=7702TO7702+M:POKEI-1,32:POKE
I,81:POKEI+FL,C:FORT=0TO20:NEXTT,I:POKE
I-1,32
 2010 FORW=ITOI+P*22STEP22:POKEW-22,32:
POKEW,81:POKEW+FL,C:NEXT:POKEW-22,32
 2020 POKEW,PEEK(A+S)+64:POKEW+FL,2:RET
URN
```

SERVIZIO SOFTWARE

Bit

Da questo numero Bit propone ai propri lettori i dischi o le cassette dei programmi pubblicati. I programmi, provati e garantiti, sono di immediato utilizzo.

Bit n°	Programma	Sistema	Prezzo	Codice	Supporto
38	Gioco della scimmia Spaccamattoni	VIC 20	15.000	VI381A	Cassetta
38	Text-Editor	PET3032	20.000	PE381B PE382B	Cassetta Disco
38	Planel	Apple II	20.000	AP382C	Disco

Per richiedere i programmi in contrassegno, pagando direttamente al postino la cifra indicata, inviare il seguente tagliando

Spedire in busta chiusa a Gruppo Editoriale Jackson - Via Rosellini, 12 - 20124 Milano

---

Inviatemi i seguenti nastri e/o dischi con i programmi pubblicati su Bit.

GRUPPO EDITORIALE JACKSON

Cod. ☐☐☐☐☐☐ **a L.** ....

Cod. ☐☐☐☐☐☐ **a L.** ....

Cod. ☐☐☐☐☐☐ **a L.** ....

Cod. ☐☐☐☐☐☐ **a L.** ....

Spese postali (contributo fisso) L. 2.000

TOTALE L. ....

che pagherò al postino alla consegna del pacco.

Cognome ....

Nome ....

Indirizzo ....

CAP .... **Città** ....

Firma ....

Seguito listato filetti e gravità

```
Lista simboli grafici

   7 : 1   SHIFT HOME =CHR$(147)

  20 : 13   CRSR↑ =CHR$(17)

  22 : 1   SHIFT HOME =CHR$(147)
       1   CRSR↑ =CHR$(17)

  23 : 2   CRSR↑ =CHR$(17)
       1   REVERSE =CHR$(18)
       1   SHIFT REVERSE =CHR$(146)
       1   CRSR↑ =CHR$(17)
       1   REVERSE =CHR$(18)
       1   SHIFT REVERSE =CHR$(146)
       1   CRSR↑ =CHR$(17)
       8   CRSR← =CHR$(29)
       1   REVERSE =CHR$(18)
       1   SHIFT REVERSE =CHR$(146)
       2   CRSR↑ =CHR$(17)

  27 : 1   CRSR↑ =CHR$(17)

  30 : 1   SHIFT HOME =CHR$(147)
       5   CRSR↑ =CHR$(17)

  41 : 1   CRSR↑ =CHR$(17)

  60 : 1   SHIFT HOME =CHR$(147)
       4   CRSR↑ =CHR$(17)

  80 : 2   CRSR← =CHR$(29)
       1   HOME =CHR$(19)

  90 : 1   HOME =CHR$(19)
       5   CRSR↑ =CHR$(17)
       17   CRSR← =CHR$(29)
       1   HOME =CHR$(19)

 680 : 1   HOME =CHR$(19)
       1   REVERSE =CHR$(18)

1000 : 1   HOME =CHR$(19)
       1   REVERSE =CHR$(18)

1020 : 1   SHIFT REVERSE =CHR$(146)

1046 : 1   HOME =CHR$(19)
       1   CRSR↑ =CHR$(17)

1080 : 1   SHIFT HOME =CHR$(147)
       3   CRSR↑ =CHR$(17)
       6   CRSR← =CHR$(29)
```

Per collaborare ... meglio!
tutti i lettori che desiderano collaborare a "Riservato Personal" devono inviare una dettagliata descrizione del programma, il relativo listato (possibilmente stampato su carta bianca) e il supporto magnetico (disco o cassetta) a:

"Riservato Personal"

Via Rosellini, 12
20124 MILANO

Sono anche disponibili le specifiche dettagliate per la preparazione degli articoli. Richiedetele allo stesso indirizzo.

a cura della **Redazione**

Un po' per problemi commerciali, un po' forse per disdegno nei confronti di un così "vile" argomento, il settore dei giochi da personal computer è di quasi totale appannaggio dei soliti americani, con qualche timida puntata giapponese. Suscita quindi un certo interesse la notizia della commercializzazione di un gioco italiano per Apple: appunto "*Avventura nel castello*". Per quanto ne sappiamo, si tratta del primo "adventure game" di concezione originale italiana.

Apriamo una parentesi per dare un'occhiata panoramica al settore dei "computer game" ed alle sue attuali linee di evoluzione. Possiamo, grosso modo (i sofisti mi perdoneranno l'approssimazione), suddividerlo in tre aree ben distinte.

Gli "arcade game"

Al primo posto nei cataloghi (e nelle vendite) stanno i soliti "arcade game", in pratica i videogiochi riportati sul personal computer. Le varianti sono innumerevoli, dall'ennesimo "invader" alla corsa automobilistica, ma sempre con due regole costanti:

1) Strategia e astuzia sono bandite o quasi: contano solo l'occhio ed i riflessi.
2) Si gioca da soli.

Indubbiamente alcuni di questi sono ben riusciti e divertenti da giocare (un esempio per tutti: Threshold della On-Line), ma l'insieme appare alquanto scialbo e ripetitivo (quante versioni di Pac-Man ci sono in giro?). Spazio per fantasia ed innovazione ce n'è dunque in abbondanza.

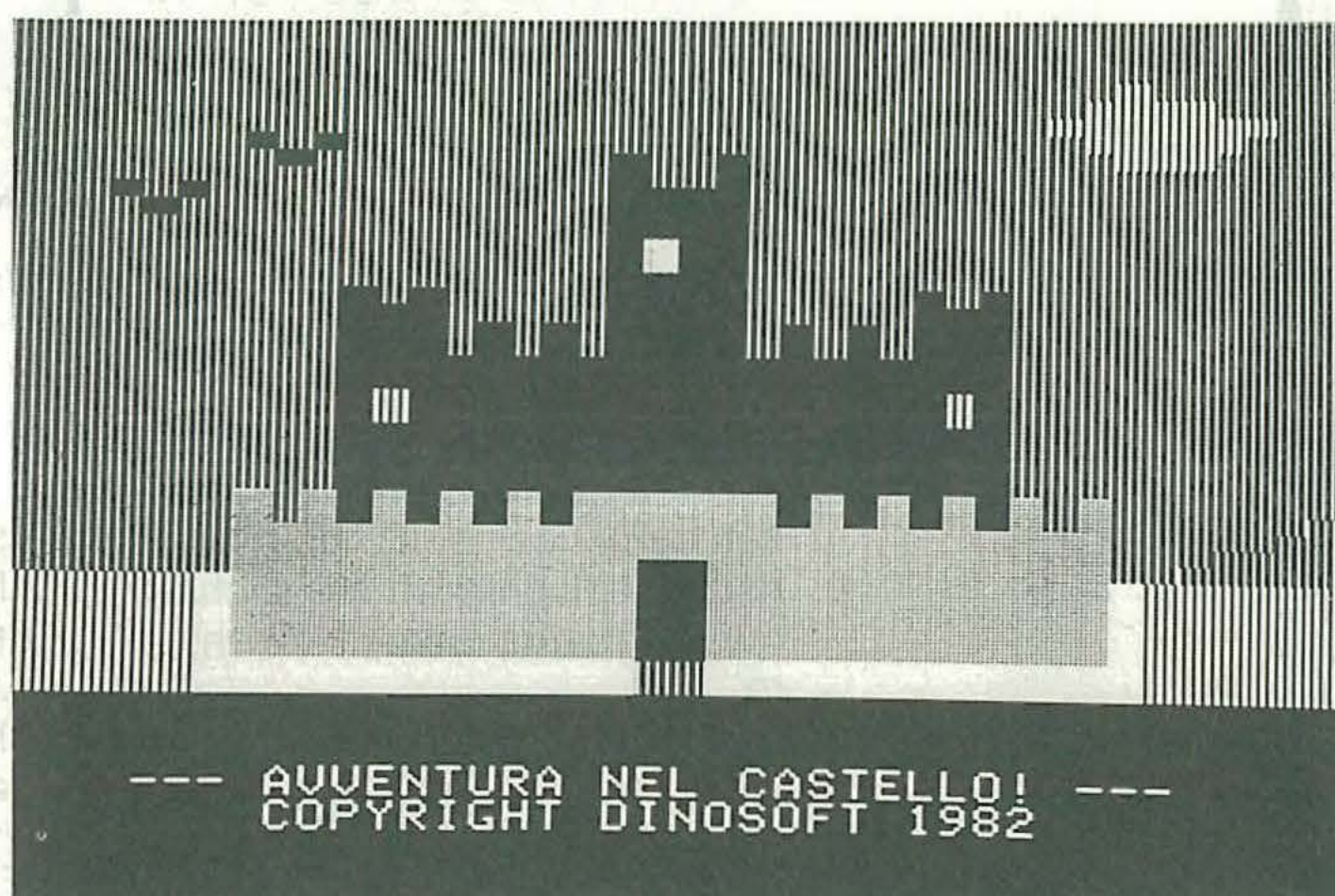
Tuttavia, come accade per certi "serial" televisivi, gli "arcade game" non possono uscire dalla loro struttura base e sono pur sempre i lontani discendenti del compianto flipper: simpatici aggeggi per rilassarsi una mezz'ora (o più...) senza pensare.

C'è stato anche qualche tentativo di realizzare giochi "tipo arcade" per più persone, ma questi tentativi sono pochissimi e per lo più mal realizzati. Tralasciamo (a malincuore) questo argomento perchè ci porterebbe troppo lontano dal discorso che oggi ci interessa.

I giochi di strategia

Ovviamente sono stati riportati sul computer tutti i più noti giochi di strategia, dal domino agli scacchi. Ben pochi sono stati invece i tentativi di realizzare qualcosa di originale sfruttando le opportunità offerte dal nuovo strumento. Anche in questo caso il copione è "tu solo contro il computer", ma si può facilmente variarlo in "noi tutti contro il computer" (e riusciremo a vincerne una, prima o poi, contro quella stupida macchina).

Il campo dei giochi di strategia è un ottimo terreno di coltura per lo sviluppo delle tecniche di intelligenza artificiale. Dal punto di vista del giocatore rimane, forse, un po' troppo arido, soprattutto per la mancanza delle

Avventura nel castello: un nuovo gioco italiano per Apple II

emozioni che sempre accompagnano il gioco tra "umani".

Gli "adventure game"

A questo punto vi aspettate: ed ecco finalmente il Gioco perfetto, che unisce tutte le migliori qualità, che sfrutta le nuove possibilità offerte dal calcolatore, ecc. ecc. Bene, non ve lo dico. Non ve lo dico perchè questo genere non esiste e, per quanto ne so, ben pochi sforzi si fanno fuori dei soliti binari, soprattutto negli USA, dove la tecnica eccellente è spesso compensata da una incredibile carenza di fantasia.
Esiste però un terzo genere di giochi, che è nato con il compu-

```
* AVVENTURA NEL CASTELLO! *

STAI PILOTANDO IL TUO MONOPOSTO SOPRA
LA DESOLATA REGIONE SCOZZESE DELLE
HIGHLANDS. HAI APPENA SORVOLATO IL
LAGO DI LOCH NESS.
IMPROVVISAMENTE IL MOTORE PERDE COLPI.
I COMANDI NON RISPONDONO!
STAI PRECIPITANDO!

COSA DEVO FARE ? ■
```

Dopo il prologo, comincia l'avventura!

```
COSA DEVO FARE ? GUARDA

SEI NELLA PIAZZA D'ARMI: UN VASTO
SPAZIO QUADRATO IN TERRA BATTUTA
CINTO DA ALTE MURA DI PIETRA GRIGIA.
AL CENTRO DELLA SPIANATA UN LASTRONE
MASSICCIO COPRE L'IMBOCCATURA DI
QUELLO CHE ERA IL POZZO DEL CASTELLO.
SENTO, IN LONTANANZA, ULULATI DI LUPI
VEDO UN PONTE LEVATOIO ALZATO.
VEDO UN PORTONE CHIUSO.

COSA DEVO FARE ? S

NON PUOI, IL PONTE E' ALZATO.

SEI NELLA PIAZZA D'ARMI.
SENTO LUPI ULULARE.
VEDO UN PONTE LEVATOIO ALZATO.
VEDO UN PORTONE CHIUSO.

COSA DEVO FARE ? ■
```

Esempio di descrizione completo, descrizione sintetica e tentativo di movimento.

```
SEI NELLA SALA DELLE COLONNE.

COSA DEVO FARE ? GUARDA

SEI IN UNA LUNGA SALA, AI LATI DELLA
QUALE SI INNALZANO DUE FILE DI ALTE
COLONNE, CHE SOSTENGONO IL SOFFITTO
ARCUATO. LE COLONNE, BENCHE' CORROSE
DAL TEMPO, PORTANO ANCORA I SEGNI DI
UNA PAZIENTE LAVORAZIONE, PER MANO
DI ABILI ARTIGIANI.
AL CENTRO DELLA SALA, UNA PICCOLA
COLONNA IN PIETRA POGGIA SU UN BASSO
PIEDESTALLO.

COSA DEVO FARE ? SPOSTA LA COLONNA

NON SI MUOVE.

COSA DEVO FARE ? ■
```

Il programma "capisce" e cerca di eseguire il comando.

```
COSA DEVO FARE ? O

SEI IN UNA STANZA CON UNA SCALA A
CHIOCCIOLA.

COSA DEVO FARE ? S

SEI NELLA CELLA DELL'ALCHIMISTA. TUT-
TO INTORNO CROGIUOLI, PESTELLI, ALAM-
BICCHI E BIZZARRI RECIPIENTI IN VETRO
DI FORMA QUANTO MAI CONTORTA. SUGLI
SCAFFALI SONO MOLTI PESANTI IN-FOLIO
DI MAGIA, ALCHIMIA, SORTILEGI. AL
CENTRO, UN TAVOLINO CHE POGGIA SU
TRE GAMBE SAGOMATE A FORMA DI ZAMPE
DI QUALCHE MOSTRUOSO ANIMALE. SUL
TAVOLINO, UN UNICO PESANTE VOLUME
RILEGATO IN CUOIO NERO:
 "L'APPRENDISTA STREGONE"

COSA DEVO FARE ? ■
```

Una particolareggiata descrizione di ambiente.

```
COSA DEVO FARE ? N

SEI NELLA LEGNAIA.
VEDO UN PICCOLO NANO CON UN
GROSSO DIAMANTE.

COSA DEVO FARE ? PRENDI UN CEPPO

NON E' POSSIBILE.

COSA DEVO FARE ? GUARDA

SEI NEL DEPOSITO DELLA LEGNA, DOVE
SONO ACCATASTATI IN PERFETTO ORDINE
RAMI SECCHI E CEPPI DI VARIE GRANDEZZ
VEDO UN PICCOLO NANO CON UN
GROSSO DIAMANTE.

COSA DEVO FARE ? ■
```

Un altro ambiente interessante.

Avventura nel castello

ter e difficilmente potrebbe esistere senza: gli "adventure game" (giochi di avventura). Sono lontani cugini dei giochi di strategia, con i quali hanno in comune la necessità di risolvere problemi. Hanno però qualcosa di originale: l'emozione della scoperta e la sfida dell'imprevisto.

Le origini degli "adventure" si perdono nella leggenda: pare che in un lontano passato (alcuni anni fa) due programmatori si siano divertiti a scrivere in FORTRAN su un DEC PDP-10 una dimostrazione della possibilità, per un calcolatore, di "capire" semplici frasi e agire di conseguenza.

Le frasi erano del tipo *GO NORTH* (vai a nord), *TAKE CHEST* (prendi il forziere) e via dicendo; lo scenario era la Colossal Cave, un'immensa caverna sotterranea colma di tesori, di pericoli e di problemi da risolvere. Il programma si chiamava (guarda caso) "Adventure".

Il programma ebbe una tale notorietà tra gli addetti ai lavori da essere presto trasportato su praticamente tutti i computer, grandi e piccoli. Ogni personal oggi ha la sua versione dell'originale "Adventure", ridotta, integrale o addirittura ampliata. Apple ne vanta ben due, praticamente identiche!

Gli "adventure", oggi

Visto il successo dell'originale, negli ultimi anni sono stati scritti tanti "adventure" da portare il genere al secondo posto nelle classifiche, dopo gli "arcade". A loro volta gli "adventure" si sono suddivisi in due filoni:

- Parlati: diretti discendenti dell'Adventure originale, in questi giochi il computer descrive in dettaglio ogni nuovo ambiente in cui si mette piede, lasciando ricreare la scena alla fantasia del giocatore-avventuriero. Esempio classico: Apple Adventure (versione Apple del primo originale).
- Grafici: gli ambienti non vengono descritti ma illustrati, meglio se in alta risoluzione ed a colori, con un brevissimo commento parlato (sempre nel senso di scritto). Esempi: il classico Mistery House e il gigantesco Time Zone (12 dischi!). Alcuni dei giochi più recenti hanno anche un minimo di animazione.

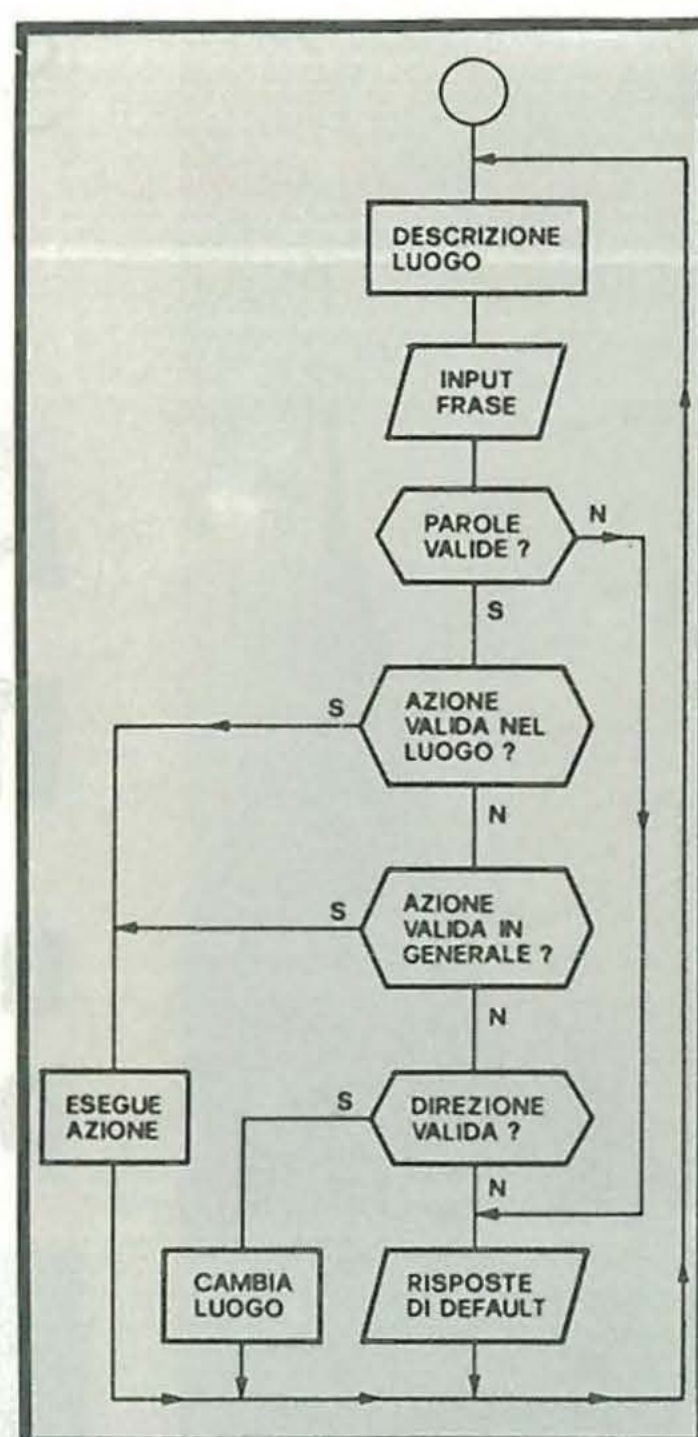



Figura 1 - Struttura del ciclo principale del programma (molto semplificata).

Parlato o grafico?

Si potrebbe pensare che gli adventure grafici abbiano messo definitivamente fuori causa quelli parlati, ma questo non è ancora avvenuto. Come qualcuno ha osservato, l'adventure grafico sta a quello parlato come il fumetto sta al racconto: un disegno è spesso meno "vero" e "reale" dell'immagine fantastica suscitata da un'abile descrizione. Inoltre la necessità di risparmiare tempo e spazio nei grafici porta in molti casi a disegni assai poco rifiniti e poveri di particolari.

A questo va aggiunto che le trame delle avventure, specialmente quelle grafiche dove l'attenzione si concentra sui disegni, sono quasi sempre inconsistenti. In genere le azioni principali sono "prendi" e "ammazza", con poca fantasia e ancor meno humour.

E in Italia?

Da noi gli adventure americani hanno ben poca diffusione, so-

prattutto per la necessità di conoscere discretamente l'inglese o meglio l'americano: giocare con il vocabolario traducendo ogni parola è piuttosto noioso.
Qualcuno ha tentato delle traduzioni in italiano, con risultati sconsolanti: occorre pensare in inglese e poi tradurre in italiano. (un esempio: "fai la doccia" diventa "prendi la doccia", dato che in inglese si dice "take shower").
Per giocare gli adventure americani ci sono dunque due alternative: aspettare traduzioni fatte sui "sorgenti" dei programmi (e qualcuno, ci dicono, sta lavorando in questo senso) o imparare l'inglese. Non scherzo: gli adventure sono un ottimo mezzo per imparare l'inglese: chi scrive ha imparato più con questo sistema che con tutti gli altri. Era comunque ora che qualcuno si decidesse a scrivere un adventure, anzi un'avventura originale in italiano.

Avventura nel castello

Hanno affrontato l'avventura (alla lettera) una coppia di bresciani, professionisti dell'elettronica industriale e inveterati giocatori di "adventure". Ne è uscito un prodotto di ottima fattura e molto divertente, specialmente se giocato con un gruppo di amici.
La trama è questa: l'aereo precipita, l'avventuriero si salva a stento in un antico castello diroccato, dove... Ma rimandiamo a **Personal Software** per la prova completa del gioco (non illudetevi di trovare anche le soluzioni).
Qui su **Bit** vogliamo invece analizzare la struttura del programma e le tecniche usate per la realizzazione su Apple II, grazie anche alle informazioni forniteci dagli autori.

Una macchina a stati

In linea teorica, sarebbe possibile scrivere un adventure come programma sequenziale, del tipo:

IF A$="APRI LA PORTA" AND LUOGO=13 AND CHIAVE=1 THEN...

QUALCHE DATO

Programma:	compilato, 24 Kbyte
Tavole e indici:	4 Kbyte
Dizionario	(tavole delle parole): 6 Kbyte Routine ausiliarie: 600 byte
Dati su disco:	26 Kbyte
Sorgenti:	28 file, per complessivi 135 Kbyte

Naturalmente si otterrebbe un programma talmente enorme da superare la capacità di qualunque personal, per non dire dell'impossibilità di scriverlo fisicamente e provarlo a fondo.
Occorre un approccio più strutturato al problema; è meglio prescindere dalla specifica avventura e realizzare una "base" universale in cui introdurre trama, mappe, oggetti e problemi dell'avventura che si vuol scrivere. Tra l'altro la base è riutilizzabile senza sforzo per ulteriori avventure.
Conviene creare quella che si dice una "macchina a stati": ogni situazione di gioco è uno "stato" tra i molti possibili, identificato dal luogo dove ci si trova, gli oggetti posseduti, le azioni effettuate, ecc.
Un' azione può non avere alcun effetto sullo stato (soltanto un messaggio di risposta) o causare un cambiamento di stato (es. *NORD* può cambiare luogo). La macchina passa dunque in un nuovo stato dove sono possibili altre azioni, che a loro volta possono determinare altri cambiamenti di stato.
Questo approccio, a prima vista complicato, consente invece, come vedremo, di semplificare notevolmente il problema.

La struttura a tavole

La macchina a stati si può realizzare con una struttura organizzata a tavole. Descriviamo una struttura di questo tipo con un esempio, riferendoci al diagramma di flusso di figura 1, che rappresenta in forma molto semplificata il ciclo principale del programma.
Per prima cosa, il programma stampa una descrizione del luogo dove ci si trova, usando la variabile *LUOGO* come indice nella tavola delle descrizioni di tutti i luoghi.
Poi aspetta la frase del giocatore, che analizza consultando le tavole delle parole (verbi, nomi, articoli, ecc.) per vedere se è composta di parole valide. In caso negativo stampa un messaggio di default e torna all'inizio.
Se le parole erano valide, consulta la tavola delle azioni valide in quel particolare luogo (ad ogni luogo è associata una tavola di frasi) per vedere se l'azione è prevista in quel luogo. Se la trova esegue l'azione (vedremo poi come) e torna all'inizio.
Altrimenti, consulta la tavola delle azioni valide dovunque (tavola unica per tutto il gioco) e, se la ricerca ha successo, esegue l'azione. Notare che la stessa azione può avere un effetto diverso in un certo luogo, nel qual caso sarebbe stata intercettata dal test precedente.
Infine, consulta la tavola delle direzioni ammesse nel luogo (che costituisce la mappa). Se coincide, cambia la variabile *LUOGO* di conseguenza, se no dà un messaggio di default.
In questo esempio mancano per semplicità gli oggetti, ma è facilmente intuibile come la struttura può essere estesa per comprenderli (tavola del luogo di ciascun oggetto, tavola degli oggetti posseduti, ecc.).
Abbiamo tralasciato di vedere cosa succede se una frase è riconosciuta valida e dev'essere eseguita l'azione corrispondente. Questa può consistere nella semplice stampa di un messaggio, oppure può implicare un cambiamento di stato; in quest'ultimo caso l'azione consisterà nel modificare la relativa variabile (es. porta aperta).

Memoria, disco e velocità

Il problema della velocità è fondamentale: se bisogna aspettare qualche secondo per avere

Avventura nel castello

la risposta, il gioco diventa subito noioso.
Per avere risposte veloci, si pensa subito a tenere tutto in memoria. Ma questo limiterebbe fortemente le dimensioni del gioco: e allora?
Avventura nel castello ha un tempo di risposta brevissimo, grazie ad un sistema ibrido: mappa, tavole ed indici vari sono permanentemente in memoria, mentre il disco contiene le descrizioni complete degli ambienti (che vengono date la prima volta) e i numerosi messaggi di risposta.
In memoria sono anche le routine di "azione" che vengono chiamate, come abbiamo visto, quando il giocatore esegue un'azione valida. I programmatori americani tendono a strutturare a tavole anche queste, costringendo però in tal modo le possibilità delle routine entro schemi fissi.
I messaggi sono registrati con la tecnica del file sequenziale indicizzato. E' un sistema che offre accesso velocissimo e massima flessibilità e di cui avremo modo di parlare in un prossimo articolo.

Tecniche e trucchi

Avventura nel castello è scritto in Applesoft... o quasi.
Se infatti l'Applesoft rimane il linguaggio di gran lunga più pratico per programmare Apple II a livello professionale, il suo limite maggiore è la bassa velocità di esecuzione. E la velocità, come abbiamo visto, è un fattore essenziale per ogni gioco.
Il programma originale, in Applesoft puro e semplice, impiegava ben otto secondi per rispondere, con pause molto più lunghe per la routine di garbage collect!
I fattori limitanti, in ordine di importanza, erano:

- la routine di garbage collect;
- il tempo di ricerca di *GOTO* e *GOSUB*;
- il tempo di accesso alle variabili;
- l'esecuzione delle istruzioni.

Una compilazione con il TASC della Microsoft avrebbe risolto tre dei problemi, lasciando inalterato il primo, il più fastidioso. Anche il TASC, infatti, impiega molto tempo a riordinare le stringhe.
Purtroppo, il TASC usa una diversa organizzazione delle variabili stringa, rendendo inutili le routine di riordino veloce, come *SPEEDFRE*, descritta in un recente articolo **Bit** n. 37 (l'avete letto, vero?).
Restava una sola, drastica soluzione: abolire le stringhe, o comunque ridurne il numero da qualche centinaio a qualche decina. Facile a dirsi. Occorreva scrivere una gestione in linguaggio macchina degli array stringa, alternativa a quella di Applesoft. Ed è proprio quello che hanno fatto gli autori di *Avventura nel castello.*

Protezione e prezzo

In Italia molti utenti (e rivenditori) hanno la pessima abitudine di far circolare copie illegali dei programmi, contribuendo a scoraggiare i già pochi autori in circolazione. I venditori di software che vogliono guadagnare il 250% hanno le loro responsabilità in questa situazione, ma questo consola ben poco chi del software vuol fare il proprio lavoro.
Avventura nel castello usa una tecnica molto efficace per scoraggiare questo genere di cose: ogni tanto, a caso e comunque dopo un po' di tempo, controlla la consistenza di alcune informazioni critiche incise sul dischetto in modo non riproducibile dai vari Locksmith & simili. Se il disco è copiato, cancella l'intera memoria. Il comportamento random rende molto lento e difficile neutralizzare la protezione, in quanto occorre molto tempo per stabilire se una copia funziona o no.
Riteniamo comunque che il prezzo di vendita decisamente basso per un prodotto di questo livello sia un argomento notevole a favore della confezione originale. ■

Avventura nel castello è in vendita presso Technoclub a L. 42.000. Per ordinazioni scrivere ai Technoclub, Via Rosellini, 12 - 20124 Milano

LIBRERIA JCE

273 Circuiti
"273 circuiti" è una raccolta di schemi per il tecnico di laboratorio e l'hobbista di elettronica. I circuiti sono tutti molto semplici e facili da realizzare. Ve n'è per tutti i gusti: per uso domestico, per autovetture, per i fissati dell'audio, per giocatori inveterati, per gli ossessionati dalle misure e dagli alimentatori, per gli appassionati di fotografia o di modellismo ecc.
Cod. 6014
L. 12.500 *(Abb. L. 11.250)*

300 Circuiti
Il libro propone una moltitudine di progetti dal più semplice al più sofisticato con particolare riferimento a circuiti per applicazioni domestiche, audio, di misura, giochi elettronici, radio, modellismo, auto e hobby.
Cod. 6009
L. 12.500 *(Abb. L. 11.250)*

Manuale pratico del Riparatore Radio TV
Il libro frutto dell'esperienza dell'autore maturata in oltre due decenni di attività come teleriparatore, è stato redatto in forma piana e sintetica per una facile consultazione.
Cod. 701P
L. 18.500 *(Abb. L. 16.650)*

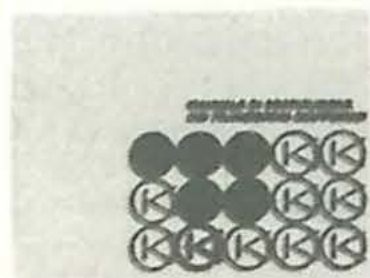
30 Programmi Basic per lo ZX 80
Programmi pronti all'uso che si rivolgono soprattutto ai non programmatori, quale valido ausilio didattico, nonchè prima implementazione del BASIC studiato, ma che possono essere, da parte dei più esperti, anche base di partenza per ulteriori elaborazioni.
Cod. 5000
L. 3.000 *(Abb. L. 2.700)*

Junior Computer Vol 1-Vol 2
Junior Computer è il microelaboratore da autocostruire su un unico circuito stampato. Il sistema base e questi libri sono l'occorrente per l'apprendimento. Prossimamente verranno pubblicati altri volumi relativi all'espandibilità del sistema.
Cod. 3001
L. 11.000 *(Abb. L. 9.900)*
Cod. 3002
L. 14.500 *(Abb. L. 11.650)*

Digit 1
Il libro mira a insegnare i concetti fondamentali di elettronica con spiegazioni semplici. Esperimenti pratici utilizzanti una piastra sperimentale a circuito stampato consentono un'introduzione graduale all'elettronica digitale.
Cod. 2000
L. 7.000 *(Abb. L. 6.300)*

Digit 2
Prosieguo naturale del Digit 1, il libro presenta oltre 500 circuiti: dal frequenzimetro al generatore di onde sinusoidali - triangolari - rettangolari; dall'impianto semaforico alla pistola luminosa.
Cod. 6011
L. 6.000 *(Abb. L. 5.400)*

Le Radiocomunicazioni
Ciò che si deve sapere sulla propagazione e ricezione delle onde em, sulle interferenze reali od immaginarie, sui radiodisturbi e loro eliminazione, sulle comunicazioni extra-terrestri ecc.
Cod. 7001
L. 7.500 *(Abb. L. 6.750)*

Manuale di Sostituzione dei Transistori Giapponesi
Il libro raccoglie circa 3000 equivalenze fra transistori giapponesi.
Cod. 6005
L. 5.000 *(Abb. L. 4.500)*

Audio & HI-FI
Una preziosa guida per chi vuole conoscere tutto sull'HI-FI.
Cod. 703D
L. 6.000 *(Abb. L. 5.400)*

Accessori per Autoveicoli
Dall'amplificatore per autoradio, all'antifurto, dall'accensione elettronica, al plurilampeggiatore di sosta, dal temporizzatore per tergicristallo ad altri ancora.
Cod. 8003
L. 6.000 *(Abb. L. 5.400)*

Le Luci Psichedeliche
Il libro descrive apparecchi psichedelici provati e collaudati, realizzazione di generatori psichedelici sino a 6 kW, flash elettronici, luci rotanti etc.
Cod. 8002
L. 4.500 *(Abb. L. 4.050)*

Il Moderno Laboratorio Elettronico
Autocostruzione di tutti gli strumenti fondamentali; alimentatori stabilizzati, multimetri digitali, generatori sinusoidali ed a onda quadra, iniettore di segnali, provatransistor, wattmetri e millivoltmetri.
Cod. 8004
L. 6.000 *(Abb. L. 5.400)*

Guida alla Sostituzione dei Semiconduttori nei TVC
Equivalenze di semiconduttori impiegati su 1200 modelli di televisori di 47 fabbricanti diversi.
Cod. 6112
L. 2.000 *(Abb. L. 1.800)*

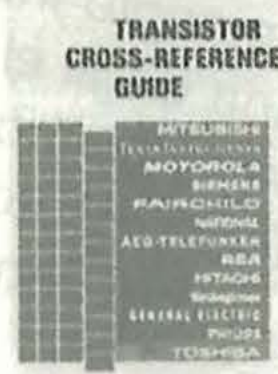
Transistor Cross-Reference Guide
Circa 5.000 equivalenze fra transistori europei, americani e giapponesi, con i relativi parametri elettrici e meccanici.
Cod. 6007
L. 8.000 *(Abb. L. 7.200)*

Costruiamo un Microelaboratore Elettronico
Per comprendere con naturalezza la filosofia dei moderni microelaboratori e imparare a programmare quasi senza accorgersene.
Cod. 3000
L. 4.000 *(Abb. L. 3.600)*

Tabelle Equivalenze Semiconduttori e Tubi Elettronici Professionali
Equivalenti Siemens di transistori, diodi, led, tubi elettronici professionali e vidicons.
Cod. 6006
L. 5.000 *(Abb. 4.500)*

Selezione di Progetti Elettronici
Una selezione di interessanti progetti pubblicati sulla rivista "Elektor". Ciò che costituisce il "trait d'union" tra le varie realizzazioni proposte e la varietà d'applicazione, l'affidabilità di funzionamento, la facilità di realizzazione, nonchè l'elevato contenuto didattico.
Cod. 6008
L. 9.000 *(Abb. L. 8.100)*

TTL IC Cross - Reference Manual
Il prontuario fornisce le equivalenze, le caratteristiche elettriche e meccaniche di pressochè tutti gli integrati TTL sinora prodotti dalle principali case mondiali, comprese quelle giapponesi.
Cod. 6010
L. 20.000 *(Abb. L. 18.000)*

Alla Ricerca dei Tesori
Il primo manuale edito in Italia che tratta la prospezione elettronica. Il libro, in oltre 110 pagine ampiamente illustrate spiega tutti i misteri di questo hobby affascinante. Dai criteri di scelta dei rivelatori, agli approcci necessari per effettuare le ricerche.
Cod. 8001
L. 6.000 *(Abb. L. 5.400)*

TV Service 100 riparazioni TV illustrate e commentate
Dalle migliaia di riparazioni che si effettuano in un moderno laboratorio TV, sono assai poche quelle che si discostano dalla normale "routine" e sono davvero gratificanti per il tecnico appassionato. Cento di queste "perle" sono state raccolte in questo libro e proposte all'attenzione di chiunque svolga per hobby o per mestiere il servizio di Assistenza TV.
Cod. 7000
L. 10.000 *(Abb. L. 9.000)*

SCONTO 10% agli abbonati

Per ordinare i volumi usare l'apposito tagliando inserito in fondo alla rivista

DataStar

di R. Stewart
e R. Massaroli

Presentazione

Se avete dei dati da archiviare in modo organico, DataStar sarà per voi un ottimo sistema di immissione e reperimento, di facile utilizzo ed elevata qualità. Le qualità più evidenti di questo pacchetto, operante sotto CP/M, sono la versatilità, la rapidità di definizione delle maschere di immissione e del formato dei dati e l'elevato grado di controllo sulla correttezza dei dati immessi.
Come è nella tradizione MicroPro, l'utente non è mai abbandonato in balia di un manuale che a prima vista può apparire pesante, ma viene guidato da menu opzionali, indicanti i comandi possibili. Sono disponibili su video in qualunque momento, a richiesta dell'operatore, spiegazioni dettagliate delle opzioni e sul significato delle domande proposte.
Il pacchetto DataStar comprende due programmi: FormGen per la definizione del formato dei dati e DataStar per la loro immissione o reperimento. I programmi utente (in BASIC, FORTRAN, COBOL ecc.) come pure gli altri componenti del sistema MicroPro (WordStar, CalcStar e SuperSort ecc.) possono facilmente attingere ai dati gestiti dal DataStar. Nel manuale sono forniti esempi di conversione di formato di dati per utilizzare con file DataStar già esistenti. DataStar è scritto in linguaggio macchina 8080, assemblato con M80 e perciò non modificabile dall'utente. Tuttavia è fornito il sorgente delle parti che interagiscono direttamente col computer, abbondantemente commentato, in modo da poter adattare il programma all'ambiente in cui opera. Per un gran numero di macchine, l'installazione è eseguibile direttamente per mezzo della utility *INSTALL*, che presenta un ampio menu di terminali e computer tra cui scegliere la personalizzazione. In caso di personalizzazione "a mano", con *DDT* o *ZSID*, per esempio, bisogna ricordare di modificare sia FormGen che DataStar.
Sul disco è fornito anche un esempio molto completo (10 file) di fatturazione (tipo USA) illustrante praticamente tutte le caratteristiche del DataStar.
Passiamo all'analisi dettagliata dei due componenti principali del pacchetto.
La utility di definizione del formato dei dati permette di realizzare sullo schermo un modulo da compilare, procedendo in modo analogo alla progettazione su carta, e definirne le modalità di immissione.

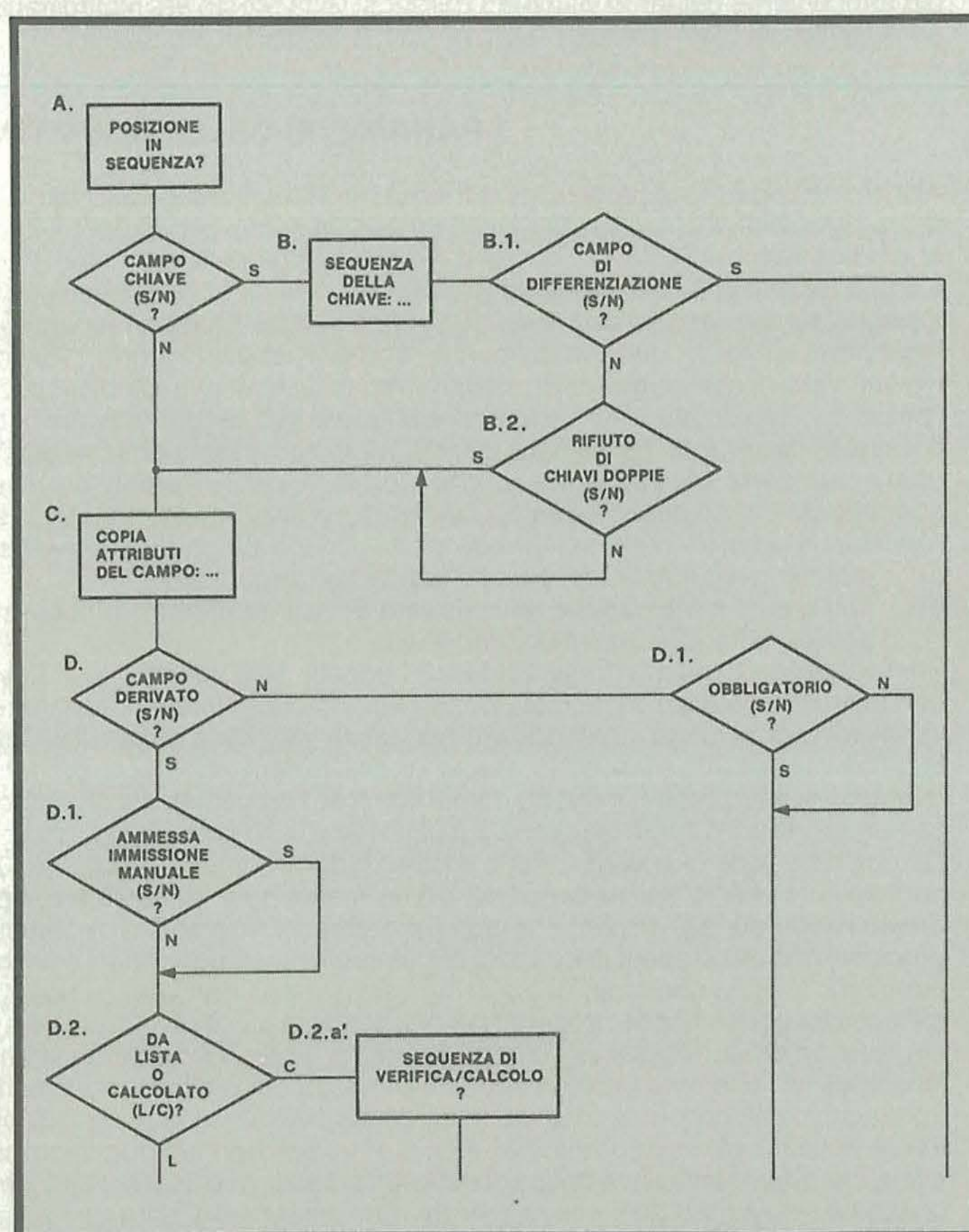
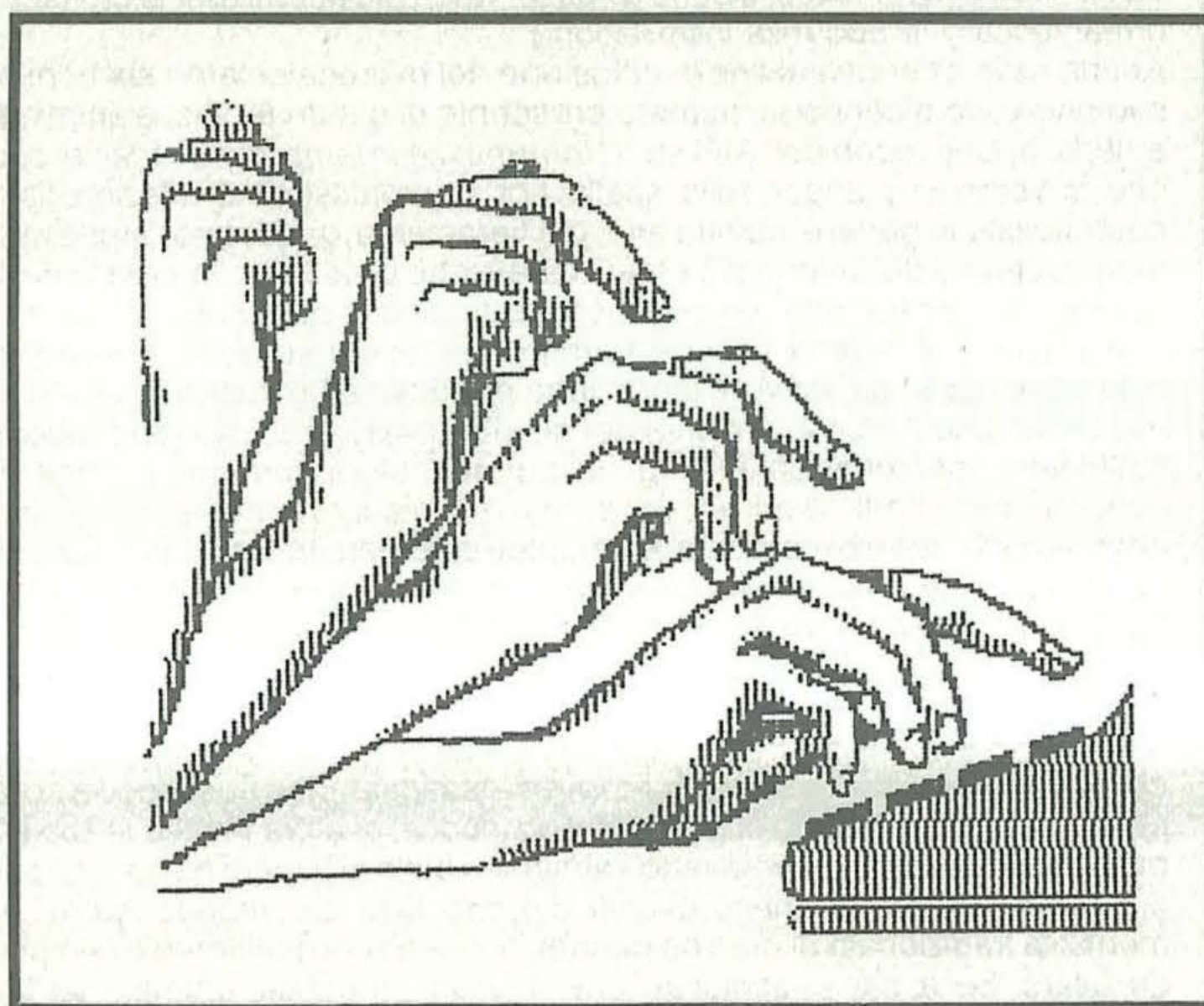



Figura 1.

Definizione del formato della maschera

Per poter creare liberamente il modulo, è necessario avere un controllo completo sul cursore. Questa capacità di editing è implementata con gli stessi comandi di agile utilizzo già noti agli utenti del WordStar.
Gli elementi che si pongono sullo schermo in questa fase sono due: campi vuoti da compilare (blank) che appaiono come sottolineature e testo esplicativo. È molto importante osservare che i blank e il testo sono del tutto indipendenti tra di loro.
Le parti descrittive (fisse e immutabili proprie di ogni modulo) possono identificare, esemplificare un blank, oppure costituire l'intestazione del modulo, fincature o spiegazioni, e allo stesso tempo i campi possono anche essere privi di commento proprio.
Ma cos'è un blank e come lo si

definisce? Il comando Q posiziona un carattere di sottolineatura in corrispondenza del cursore. Ogni sequenza consecutiva di ˆQ forma un blank. I dati vengono organizzati in un file, uno per ogni applicazione. Il file non è altro che l'insieme di tutte le informazioni, diviso in tanti record che corrispondono a tutto un sottoinsieme di informazioni (1 per ogni singolo modulo interamente compilato) e ogni blank è la visualizzazione di un campo vuoto di un record. Non è necessario limitarsi alla grandezza dello schermo, perchè un modulo può estendersi e arrivare in lunghezza o larghezza fino a 255 caratteri.
Nonostante il programma venga fornito con un esauriente manuale, l'utente smaliziato può cominciare a lavorare, sfruttando i menu help. Stiamo entrando nell'ambito del ˆJ, comando che seleziona quattro livelli di aiuto, disponibili in qualunque momento. Come è intuibile, tutti i comandi diretti al programma sono costituiti da un carattere solo preceduto da *CTRL*.
Basilare, per l'identificazione di un record è la chiave. DataStar obbliga l'utente a definire uno o più campi come chiave, per costruire un file di indice, che servirà per rendere più veloce l'accesso alle informazioni. Questa operazione si effettua posizionando il cursore sul campo e premendo ˆK.

Definizione dei campi

Sebbene sia sufficiente disporre il testo e i campi per poter passare alla fase di immissione, si sfruttano appieno le capacità del DataStar solo definendo la natura dei dati accettabili in fase di immissione e le modalità dell'immissione stessa.
Posizionando il cursore su un blank e premendo ˆR, viene proposta una serie di opzioni e caratteristiche attribuibili ai campi, (tabella 1) analizzate e schematizzate nel flowchart.
Durante la fase di definizione della maschera è possibile dare il comando ˆW, per passare alla stampa di tabulati, rispecchianti vari aspetti del modulo definito; per poter verificare a

USER FRIENDLY (PACKAGE E PROBLEMATICHE D'UTENTE)

Con questa recensione del DataStar **Bit** intende iniziare nel modo più sistematico possibile la disamina dei più significativi prodotti-programma, specie del tipo "general purpose", il cui numero va crescendo e la cui qualità di pari passo va migliorando, a testimonianza dell'importanza decisiva che sull'arena dei personal proprio questi oggetti hanno avuto e continuano ad avere per la loro prorompente diffusione.
DataStar si riconnette idealmente a quel WordStar che fu al centro - se non altro per il suo carattere di best seller - dello speciale sul word processing (**Bit** n. 29/1982) e già allora accennammo agli altri package (MailMerge, CalcStar ecc.) insieme ai quali forma una rosa di elementi correlati su cui la MicroPro fonda la sua fortuna commerciale. Ma la continuità che si intede d'ora in poi assicurare a questi discorsi deriva principalmente dalla constatazione che la materia è oggi più che matura.
Facciamo a tale proposito soltanto la considerazione che il successo dei PC deriva moltissimo da simili prodotti il che è dimostrato non solo dalla scarsa diffusione di macchine il cui hardware pure è validissimo ma anche dalla fortuna più rapida del previsto che taluni nuovi microsistemi a 16 bit vanno riscuotendo benchè tutt'ora meno forniti di applicativi specifici: la scelta di vestirli subito con VisiCalc, sistemi di gestione di basi di dati e simili è stata felice, dicono gli osservatori più attenti (per la cronaca, VisiCalc, secondo alcune stime ha ormai toccato le 500 mila installazioni).
Anche nelle aziende italiane la diffusione dei microcalcolatori sta cominciando a dilagare: lo voglio qui, sul mio tavolo, va dicendo un numero crescente di quadri tecnici e amministrativi e la cosa (secondo quanto è emerso al Convegno dell'AIPI su *"L'Informatica in tempi di povertà* nel corso del quale chi scrive manca poco che si vedesse pianger sulla spalla fior di professionisti affezionatissimi al COBOL, al batch e all'EDP centralizzato in genere, mentre altri, più fieramente, gridavano "morte al personal!"... o giù di lì) impensierisce molti sacerdoti dell'informatica tradizionale che paventano la perdita del loro potere di controllo di quell'utente finale da loro finora così sdegnosamente (anche per cause di forza maggiore, siamo giusti) trascurato se non velipeso. Merita la pena di fermarsi un attimo su questo emergente fenomeno, senza la pretesa di esprimere pareri definitivi nè tantomeno perfettamente documentati. Ci sembra però di poter asserire che si assiste ad una (tardiva) rincorsa del disaffezionato end user cercando di riconquistarlo con le blandizie del cosiddetto "personal computing" la cui sfumatura semantica come ebbi modo di accennare tempo fa, consiste nell'offerta degli stessi servizi (di elaborazione testi, calcolo personale, gestione e recupero informazioni) ottenibili sui microcomputer però tramite terminali connessi a filo doppio al mainframe, la cui centralità viene così ribadita, confermandone la religione. Lungi da noi, in queste poche righe, ogni idea di totale partigianeria pro-personal, il cui carattere attualmente "anarchico" non va sottaciuto, insieme ai seri problemi che ciò può comportare in una corretta gestione concertata (o, come si dice oggi in informatica, "integrata" dell'azienda e dei suoi dati). Tuttavia molti fattori rendono assai problematica la riconquista del perduto bene con simili mezzi: se infatti si tratta di prodotti "casereccі" essi hanno quasi la stessa probabilità che in passato di rivelarsi intempestivi ed inadeguati, non riuscendo a competere con prodotti "di massa" che però vantano anni-uomo di R. & S. sul groppone; qualora invece si abbia a che fare con package studiati per mini o megacomputer non di rado - almeno a detta di esperti da noi interpellati in più occasioni - quando non si dimostrino addirittura meno evoluti dei loro rivali del mondo micro, hanno un impianto che tradisce la mentalità tradizionale di chi li ha concepiti, ossia sono inutilmente complicati e scarsamente facili da capire e da usare. Se a ciò si aggiunge che il costo (a causa, ahimè, del loro carattere "nobile", meglio della ristrettezza del loro mercato) è spesso non lieve e tende così a prosciugare i budget dei singoli reparti molto più atrocemente dei tanto vituperati personal, si fa presto ad immaginare una tragica inutilità degli sforzi dei pur volonterosi EDP manager (salvo errori & omissioni, beninteso).

I PARAMETRI DELLA SIMPATIA

Alla fin fine la carta vincente (specialmente per la marea degli utilizzatori singoli: professionisti e microsocietà, che problemi di complessità aziendale non he hanno proprio, beati loro) è il basso costo, che questi tempi di povertà rendono solo più decisivo. Però i package dei personal sovente possiedono anche il fascino, condito di praticità, dell'ergonomia e della maggior facilità d'impiego e occhieggiano all'utente con le lusinghe del loro esser "user friendly", come si dice. Questo fenomeno talora assume aspetti di blandizie incredibile, al limite dell'adescamento, come in quella Monna LISA da Apple ultimamente presentata. Notiamo qui, di passaggio, che qualcuno l'ha definita uno (splendido) pezzo di software (e a noi latini questo "pezzo di..." associato al nome femmineo finisce per suggerire erotiche connotazioni). A questo punto però occorre andarci cauti, con la propaganda. Mi spiego: essere amici verso l'utente ha un senso ben diverso per una rivista seria rispetto a quello che possiede per un produttore o un commerciante. In altri termini, riconosciuta anche noi come universalmente avviene, l'importanza di un simile fattore - tecnicamente si dice *l'interfaccia utente* - resta sempre da porsi almeno un paio di domande:
a) quanto, ovviamente, essa risulti valida nei singoli casi;
b) fino a qual punto, anche da un punto di vista generale, un eccesso di "amichevolezza" non rischi di essere, oltre che superfluo, dannoso.

Per far intendere questo (forse contorto) concetto farò riferimento al VisiCalc, tanto se ne rilevo una pecca non toglierò nulla alla sua meritata popolarità. Notoriamente è una matrice sterminata di caselle di ampiezza variabile, di gran lunga superiore alle necessità reali della quasi totalità degli utenti. A parte il fatto che, se a qualche mentecatto venisse in mente di utilizzarne anche solo la metà, l'usuale spazio dei 64 Kbyte di memoria si saturerebbe, viene da domandarsi se i progettisti non avrebbero fatto meglio a rinunciare a tanta spettacolarità a favore di faccende più utili, come la sintassi delle formule più conforme all'algebra ordinaria. Questa riflessione ci collega al serio problema dei *parametri* da definire ai fini di una corretta valutazione di un package. Poiché lo spazio a questo punto tiranneggia rinviamo il rompicapo a momenti successivi, con l'invito rivolto sia agli esperti che agli utilizzatori ad intervenire. A chiusura di queste noterelle, dopo aver precisato che dei prodotti poco validi ovviamente non si parlerà (il che non esclude certo che ve ne siano, ma nemmeno implica che quelli di cui al momento si tace non vadano bene), comincerò col sottolineare che la difficoltà maggiore riposa nel carattere soggettivo e *qualitativo* di tali parametri. Per sovrammercato concetti come ergonomia, interfaccia utente, tempo di apprendimento e potenza *non* sono affatto mutuamente indipendenti: l'esempio già fatto sopra lo dimostra ma è altrettanto intuitivo che la rapidità d'apprendimento contrapposta alla potenza costituisce un'alternativa di non facile risoluzione, se non nei prodotti più spartani. Senza fantasticare circa possibili analogie di prodotti banda x guadagno su questo scivoloso terreno diciamo allora che la presentazione critica di concreti prodotti, che fin d'ora affidiamo a gente che li conosce a dovere, contiamo possa gradualmente migliorare man mano che queste idee, col progresso dell'arte, oltre che con la pratica d'uso, andranno affinandosi.

G.G.

```
TEST FORM LISTING AND FIELD ATTRIBUTE DEFINITIONS

                  FIELD NUMBERS

----------------------------------------------------------------------
i                      DITTA TAL DEI TALI S.p.A.                     i
i                                                                    i
i                          Anagrafica clienti                        i
----------------------------------------------------------------------

  NOME       ____________________1          CODICE FISCALE   ______________8
  INDIRIZZO  ____________________2
  CITTA'     __________3 CAP ____4          CREDITO ______9
                                            DEBITO  ______10
                                            SALDO   ______11 in DATA 12/_13/14
                                       NOTE: ______________________________15
  TELEFONO   ____5/_________6                  ______________________________16
  TELEX      ______________7                   ______________________________17

---> N.B. Per CAP e prefissi telef. ci sono gli elenchi nel cassetto destro.
```

```
TEST FORM LISTING AND FIELD ATTRIBUTE DEFINITIONS

              RANGE CHECK, LOW

----------------------------------------------------------------------
i                      DITTA TAL DEI TALI S.p.A.                     i
i                                                                    i
i                          Anagrafica clienti                        i
----------------------------------------------------------------------

  NOME       ____________________           CODICE FISCALE   _______________
  INDIRIZZO  ____________________
  CITTA'     ___________ CAP _____          CREDITO ________
                                            DEBITO  ________
                                            SALDO   ________ in DATA 0_/___/83
                                       NOTE: ______________________________
  TELEFONO   _____/__________                  ______________________________
  TELEX      ______________                    ______________________________

---> N.B. Per CAP e prefissi telef. ci sono gli elenchi nel cassetto destro.

              RANGE CHECK, HIGH

----------------------------------------------------------------------
i                      DITTA TAL DEI TALI S.p.A.                     i
i                                                                    i
i                          Anagrafica clienti                        i
----------------------------------------------------------------------

  NOME       ____________________           CODICE FISCALE   _______________
  INDIRIZZO  ____________________
  CITTA'     ___________ CAP _____          CREDITO ________
                                            DEBITO  ________
                                            SALDO   ________ in DATA 31/___/99
                                       NOTE: ______________________________
  TELEFONO   _____/__________                  ______________________________
  TELEX      ______________                    ______________________________

---> N.B. Per CAP e prefissi telef. ci sono gli elenchi nel cassetto destro.


TEST FORM LISTING AND FIELD ATTRIBUTE DEFINITIONS

              ENTRY CONTROL MASK

----------------------------------------------------------------------
i                      DITTA TAL DEI TALI S.p.A.                     i
i                                                                    i
i                          Anagrafica clienti                        i
----------------------------------------------------------------------

  NOME       ____________________           CODICE FISCALE   _______________
  INDIRIZZO  ____________________
  CITTA'     ___________ CAP _____          CREDITO ________
                                            DEBITO  ________
                                            SALDO   ________ in DATA __/___/__
                                       NOTE: ______________________________
  TELEFONO   _____/__________                  ______________________________
  TELEX      ______________                    ______________________________

---> N.B. Per CAP e prefissi telef. ci sono gli elenchi nel cassetto destro.

              CONTENT CONTROL MASK

----------------------------------------------------------------------
i                      DITTA TAL DEI TALI S.p.A.                     i
i                                                                    i
i                          Anagrafica clienti                        i
----------------------------------------------------------------------

  NOME       ____________________           CODICE FISCALE   CCCCCC99C99C999C
  INDIRIZZO  ____________________
  CITTA'     ___________ CAP _____          CREDITO 99999999
                                            DEBITO  99999999
                                            SALDO   ________ in DATA 99/CCC/99
                                       NOTE: ______________________________
  TELEFONO   99999/9999999999                  ______________________________
  TELEX      ______________                    ______________________________

---> N.B. Per CAP e prefissi telef. ci sono gli elenchi nel cassetto destro.


TEST FORM LISTING AND FIELD ATTRIBUTE DEFINITIONS

                        FIELD ATTRIBUTE DEFINITIONS

                        Q=required
                        C=check dgt
                        J=right just      * * D E R I V E D *
                        W=write ed c      .     LIST     CALC ******VERIFICATION****
                        O=oper entry      .                   .
                        R=range chk  PAD/ . INDEX ITEM        .           LIST VERIFY
FLD LEN LIN COL KEY E=edit mask FLOAT . FIELD  NUM ORDER .    ORDER   FILE NAME
                    .      .          .                   .
001 021 006 013 001 . Q      . P      .                   .
002 021 007 013 002 .        . P      .                   .
003 011 008 013     .        . P      .                   .
004 005 008 029     .        . P      .                   . S  001
005 005 013 013     .   J  E . P  F0  .                   . R  002
006 010 013 019     .      E . P      .                   . R  003
007 016 014 013     .        . P      .                   . R  004
008 016 006 062     .      E . P      .                   . R  005
009 008 008 052     .   J  E . P      .                   .
010 008 009 052     .   J  E . P  F-  .                   .
011 008 010 052     .   J    . P      .            006 N .
012 002 010 069     .   J RE . P      .                   .
013 003 010 072     .      E . P      .                   . L  007   A:MESI
014 002 010 076     .     RE . P      .                   .
015 032 012 046     .        . P      .                   .
016 032 013 046     .        . P      .                   .
017 032 014 046     .        . P      .                   .



                        CALCULATIONS

#011=#9+#10
```

Listato 1 - Tabulati di verifica della definizione dei campi del FormGen.

colpo d'occhio le caratteristiche principali:

Stampa 1: sequenza dei campi.
Stampa 2: controlli campo di variabilità valori minimi, valori massimi.
Stampa 3: maschera di controllo di immissione, contenuto.
Stampa 4: attributi dei campi, derivazioni e verifiche.
Stampa 5: calcoli.

Definita la disposizione del testo e dei campi della maschera e le modalità di immissione, si può terminare questa fase preparativa, digitando ^C. Se a questo punto sono presenti errori, mancanze o incongruenze nella definizione della maschera, tali errori vengono segnalati. La fase FormGen terminerà soltanto dopo aver posto rimedio.
Se invece non ci sono problemi è possibile passare all'immissione con DataStar dando il comando *D* o salvare il modulo e tornare al sistema operativo con il comando *B*.
La logica di funzionamento del DataStar, almeno per quanto riguarda l'immissione, dovrebbe essere già molto chiara alla luce di quanto illustrato riguardo al FormGen e alle caratteristiche attribuibili ai vari campi.

I comandi di editing

Nel corso dell'utilizzo del DataStar sono disponibili vari comandi di editing. I comandi di spostamento del cursore, sono molto simili a quelli del WordStar, ovvero ^*S* e ^*D* rispettivamente per spostare a sinistra e a destra di un carattere il cursore e ^*A* ^*F* per posizionarlo sul campo precedente o successivo. Troviamo inoltre ^*G* per cancellare il carattere su cui è posizionato il cursore, *DELETE* o *RUB* o la freccia indietro (a seconda della tastiera utilizzata) per cancellare il carattere precedente e ^*V* per inserire uno spazio davanti al cursore. Altri comandi sempre disponibili sono: ^*U* e ^*O* per la stampa del modulo visualizzato, rispettivamente con e senza il testo descrittivo. ^*U* serve nei casi in cui non si utilizza un modulo prestampato su carta, ^*O* invece quando si desidera compilare su carta un modulo prefincato.
^*Z* ripristina la videata. È utile se per cause varie la viedata viene cancellata (distacco del terminale per esempio) o sporcata con caratteri spurii (elettricità statica o connessioni ballanti) o a causa di messaggi provenienti dal sistema operativo (*BDOS ERR ON A:.... o DISK WRITE RETRY*, per esempio). Ricostruisce tutta la videata (ammesso che la causa non abbia influito sul regolare funzionamento di DataStar) lasciando intatti i dati immessi fino a quel momento.
^*J* abilita e disabilita le funzioni di *HELP*, ovvero i menu e le spiegazioni automatiche. È particolarmente vantaggioso quando si usano moduli che superano la lunghezza della parte libera di schermo (generalmente 17 righe), per poter sfruttare le altre sei righe altrimenti occupate dai menu.

La struttura del DataStar

Le opzioni del DataStar si ramificano tutte da un menu principale detto *MODE SELECTOR MENU*, e sono le seguenti:

Figura 5 - Mode selector menù.

Figura 4.

Figura 3

Figura 2

Aggiunta: aggiunta di nuovi record.
Reperimento: selezione perchiave.scansione in sequenza del file indice, scansione in sequenza del file dati, modifica maschera di scansione.
Batch: selezione uso file batch verifica di batch.
Varie: manutenzione del file aiuto uscita dal modulo.

Aggiunta di record

Questa opzione consente all'utente di immettere nuovi gruppi di dati (record) secondo le specifiche definite per tale modulo usando FormGen. Naturalmente in questa fase verranno archiviati solo i dati e non il testo esplicativo. Nel corso della immissione, il cursore può essere spostato solo entro i campi appositi. Comando specifico della fase di immissione è ^C, usato per copiare il contenuto dello stesso campo della precedente

immissione. È molto comodo nel caso di immissione di dati in cui alcuni campi sono ripetitivi. L'immissione sarà fedelmente guidata e controllata secondo le vostre specifiche. Durante l'inserimento avvengono già alcuni controlli diretti sulla legalità dei caratteri immessi.
Terminata la compilazione del modulo, (tramite ^B o uscendo dal campo ultimo della sequenza), inizia la fase di controllo, verifica e calcolo. Viene controllato che laddove l'immissione è stata definita obbligatoria, quei campi o caratteri vengano effettivamente immessi. Il valore dei campi a controllo di campo di variabilità viene confrontato con i valori limite. Intervallati dalle verifiche possono essere effettuati i calcoli e visualizzati i relativi risultati.
In questa fase (come nel resto del programma) i messaggi di errore visualizzati scompaiono solo dopo la pressione di un particolare tasto, corretta impostazione della visualizzazione dei messaggi, che troppo spesso, anche in software di un certo livello, scompaiono automaticamente senza dare all'utente il tempo di leggerli.
Se tuttavia è stato selezionato il modo *BATCH* (vedi paragrafo apposito) o se non ci sono calcoli e verifiche, la fase di verifica viene saltata e i dati registrati così come sono.
Al termine viene dato all'operatore un'ultima occasione per rettificare ciò che ha immesso: in risposta all'apposito messaggio può rispondere con il *RETURN* per archiviare i dati e passare a un'altra immissione o premendo *DELETE, RUB*, o la freccia indietro (a seconda della tastiera usata), cancellare i dati immessi e ricominciare.

Reperimento dei dati

Esistono tre modi di ricerca di dati immessi:
- per chiave;
- per scansione in ordine di indice;
- per scansione in ordine del file dati.

Una volta reperito un record è

Tabella 1 Definizione dei campi.

A. POSIZIONE IN SEQUENZA: ——— [FIELD ORDER: ———]

Consente di specificare la sequenza con cui verranno richiesti i dati durante la fase di immissione, indipendentemente dalla posizione sullo schermo, fattore che si riflette nell'ordine con cui verranno scritti sul file. Viene in pratica definito un numero di identificazione del campo.
Default: Numerazione dall'alto in basso e da sinistra verso destra secondo la disposizione.

B. SEQUENZA DELLA CHIAVE: ——— [KEY ORDER: ———]

Ha rilevanza solo se il campo in questione è stato specificato come uno dei campi chiave. Definisce l'ordinamento del file per consentire il reperimento rapido.
Default: L'ordine con cui i campi sono stati definiti chiavi.

B.1. CAMPO DI DIFFERENZIAZIONE? (S/N) [TIE BREAKER FIELD? (Y/N)]

Questa domanda viene proposta solo per l'ultima di più chiavi. Rispondendo affermativamente, DataStar immetterà in fase di immissione automaticamente il valore numerico più basso necessario a rendere unica la chiave, in caso di chiavi uguali.
Default: No.

B.2. RIFIUTO DI CHIAVI DOPPIE? (S/N) [REFUSE DUPLICATE KEYS (Y/N)]

L'uso di questa opzione rende impossibile l'immissione di chiavi uguali tra di loro.
Default: Yes.

C. COPIA ATTRIBUTI DEL CAMPO: ——— [COPY ATTRIBUTES OF FIELD: ———]

Serve nel caso in cui più campi hanno la medesima definizione, per abbreviare il lavoro di definizione del campo. Es.: i tre campi di *NOTE* in basso a destra dell'esempio hanno gli attributi uguali.
Default: Il numero del campo stesso.

D. CAMPO DERIVATO? (S/N) [FIELD DERIVED? (Y/N)]

Un campo può venir immesso manualmente o reperito in uno dei due modi automatici: tramite calcolo sulla base di campi precedenti o prelevando il dato da un altro archivio. Il calcolo inoltre può essere matematico o un concatenamento di campi alfanumerici. Es.: il campo *SALDO* dell'esempio è la somma di *CREDITO* e *DEBITO.*
Default: No.

D.1. AMMESSA IMMISSIONE MANUALE? (S/N) [ALLOW OPERATOR ENTRY? (Y/N)]

Consente la correzione manuale dei campi calcolati o derivati.
Default: No.

D.2. DA LISTA O CALCOLATO? (L/C) [LIST/CALCULATED? (L/C)]

Un campo derivato da lista utilizza un altro campo del modulo come indice in un altro file, per prelevare il proprio valore. Un campo calcolato è il risultato di una formula che ha come operandi il valore di altri campi dello stesso modulo e operatori numerici o di stringa.
Default: L.

D.2.a. CAMPO INDICE IN LISTA NUMERO: ——— [INDEX FIELD NUMBER: ———]

Specifica il numero del campo del modulo che viene utilizzato come indice per prelevare il valore del campo in oggetto.
Nessun default.

D.2.b NUMERO DEL CAMPO NELLA LISTA: ——— [ITEM NUMBER IN LIST:———]

Determina all'interno del record individuato dal campo indice (D.2.a), da quale campo prelevare il valore del campo in oggetto.
Default: 001.

D.2.a'. ORDINE DI VERIFICA-CALCOLO [VERIFY/CALCULATE ORDER:]

Determina l'ordine con cui vengono calcolati e verificati i campi. In generale, ha rilevanza solo in casi in cui i campi vengono derivati sulla base di altri campi derivati. In tali casi è bene specificare che la verifica del risultato della prima derivazione/calcolo preceda la seconda derivazione/calcolo. Es.: se L. 009 = L.5+L. 6 e a sua volta L. 005 = L. 1 + L. 3 occorre verificare L. 5 prima del calcolo di L. 9.
Default: 001.

D.2.b' CALCOLO NUMERICO O STRINGA? (N/S)? [NUMERIC/STRING? (N/S)]

I calcoli possono essere numerici o per concatenamento di campi alfanumerici. I calcoli numerici possono comprendere le quattro operazioni fondamentali e l'esponenziazione e le parentesi. Per quanto riguarda i campi alfanumerici, l'unica operazione possibile è il concatenamento di campi interi, parti di campi e costanti.
Default: N.

D.2.b'i. ESPRESSIONE ALFANUMERICA PER IL CAMPO #000=
[ENTER STRING EXPRESSION FOR FIELD: #000=]

L'espressione è costituita dagli indicatori dei campi (il relativo numero preceduto da L.) parti di campi e costanti concatenate con l'operatore]. Le parti di campi si definiscono facendo seguire all'indicatore di campo una parentesi contenente due numeri separati da virgola. Il primo di questi specifica il carattere di inizio della parte, mentre il secondo indica il numero di caratteri dal primo che interessano. Laddove il secondo numero è mancante si intende fino a fine campo. Nella espressione possono comparire inoltre costanti, ovvero gruppi di caratteri, i quali vanno racchiusi tra virgolette (").
Nessun default.

D.2.b'ii. ESPRESSIONE ALGEBRICA PER IL CAMPO #000=
[ENTER ALGEBRIC EXPRESSION FOR FIELD: #000=]

Come per le espressioni alfanumeriche, vengono utilizzate gli indicatori di campo (Es.: L. 008). Le operazioni riconosciute sono +,-,*,/ e ^ per l'esponenziazione. Sono ammessi costanti e parentesi.
Nessun default

consentito stamparlo, modificarlo con gli stessi comandi utilizzati durante l'immissione, oppure cancellarlo del tutto. Dopo eventuali modifiche, i campi vengono verificati se necessario.

Ricerca per chiave

Questo metodo di ricerca consente il rapido reperimento di singoli record. Immettendo la chiave cercata (Es.: *NOME*), DataStar ricerca il record relativo (sfruttando l'indice) e se esiste ne visualizza l'intero contenuto. Questo modo di accesso è rapido in quanto lavora sul file indice, ricercando per bisezione in modo da localizzare la chiave a cui corrisponde un puntatore al record dati relativo. La chiave consiste di tutti i campi (molto spesso uno solo) disegnati come chiavi durante la definizione del modulo con FormGen.

Ricerche per scansione

Le ricerche per scansione sono soggette alla definizione di una maschera di ricerca, che fissa i parametri di selezione della ricerca. Questa maschera è di formato identico a quello di immissione, con la differenza che tutte le posizioni di tutti i campi sono riempiti con asterischi. Modificando il contenuto di parte dei campi, si ottiene che in fase di scansione (per indice o in ordine di file dati), vengano visualizzati solo quei record che rispondono ai requisiti posti. Se ad esempio in un file di anagrafiche si vogliono ricercare tutti i nominativi in provincia di Como, si deve sostituire ai due asterischi nel campo della sigla della provincia, i caratteri CO, e premere ^*B*, passando automaticamente in fase di scansione per indice, visualizzando uno dopo l'altro esclusivamente i record aventi la sigla di provincia di CO. Per la compilazione e la modifica della maschera di scansione valgono gli stessi comandi che in fase di immissione di nuovi record. Le definizioni date alla maschera di scansione rimangono valide per tutte le ricerche, fino ad ulteriore ridefinizione.

D.1'. OBBLIGATORIO? (S/N) [REQUIRED? (Y/N)]

La risposta negativa consente all'operatore di dare un input nullo *(CR)*.
Default: N.

E. ALLINEAMENTO A DESTRA? (S/N) [RIGHT JUSTIFY? (Y/N)]

Determina su quale estremità verrà allineato il contenuto del campo se non riempito completamente.
Default: N.

F. RIEMPIMENTO PARTE VUOTA? (S/N) [PAD FIELD? (Y/N)]

Permette il riempimento tramite un carattere definibile dall'utente delle parti di un blank rimaste inutilizzate. Es.: La parte libera di un campo numerico contenente un importo può essere riempita con asterischi per prevenirne l'alterazione dopo la stampa.
Default: N.

F.1. QUALE CARATTERE PER RIEMPIMENTO? [ENTER PAD CHARACTER:]

Serve per definire il carattere con cui va riempita la parte vuota del campo. Caratteri tipici sono spazio, asterisco, zero.
Default: Nessun default.

G. CARATTERE VOLANTE? (S/N) [FLOATING CHARACTER? (Y/N)]

Un carattere "volante" viene inserito automaticamente nel campo, a sinistra dei dati se essi sono allineati a destra, o viceversa. Può essere utile per il segno di valuta, lo zero dei prefissi telefonici, +, -, ecc..
Default: N.

G.1. QUALE CARATTERE VOLANTE? [ENTER FLOATING CHARACTER:]

Definisce il carattere anteposto o postposto al contenuto del campo.
Default: Nessun default.

H. VERIFICA DEL CAMPO? (S/N) [VERIFY FIELD? (Y/N)]

Dando risposta affermativa a questa opzione, il campo verrà controllato dopo la sua immissione. La verifica può essere di tre tipi: a vista (sight), per ribattitura (retype), per confronto con una lista (list). La verifica può essere effettuata modulo per modulo o in batch.
Default: N.

H.1. VERIFICA DI TIPO VISTA/RIBATTITURA/LISTA? (V/R/L) [SIGHT/RETYPE/LIST? (S/R/L)]

Questa domanda consente di scegliere il tipo di verifica desiderata.
La verifica a vista, a modulo completato, posiziona il cursore su ciascuno dei campi per cui è richiesta la verifica, attendendo la conferma con *CR* o permettendo l'editing del campo.
La verifica per ribattitura è analoga, salvo che il campo in questione viene pulito e l'utente deve ribattere esattamente il contenuto del campo. In caso di non corrispondenza tra le immissioni, il campo viene riproposto fino ad ottenere due immissioni successive identiche.
La verifica per confronto con lista ricerca il dato immesso, in un elenco di dati validi, accettandolo solo se presente nell'elenco stesso. L'elenco può risiedere su disco o in memoria durante la fase di immissione.
Default: S.

H.1.a. TENERE IL FILE IN MEMORIA DURANTE L'IMMISSIONE? (S/N) [KEEP FILE IN MEMORY DURING DATA ENTRY? (Y/N)]

È possibile, durante l'immissione, tenere il file di verifica in memoria. Si ha il vantaggio di una maggiore velocità di verifica (fattore non indifferente in sistemi con dischi lenti). La grandezza massima del file non deve però superare 500 caratteri.
Default: N.

H.1.b. NOME DEL FILE DI VERIFICA? [ENTER LIST FILE NAME:]

Qui si specifica il nome del file di verifica, evidentemente entro le specifiche dei nomi di file sotto CP/M.
Nessun default.

H.1.c. DRIVE DEL FILE DI VERIFICA? (—/A/B...) [ENTER LIST FILE DISK DRIVE (—/A/B/....)]

I drive validi sono A-P.
Default: (drive selezionato).

H.1.d. CAMPO NEL FILE DI VERIFICA? [ITEM NUMBER OF FIELD?]

Specifica il campo del record del file di verifica col quale confrontare il dato immesso.
Default: 001.

H.1.a'. VERIFICA BATCH? (S/N) [BATCH VERIFY? (Y/N)]

La risposta affermativa fa sì che i moduli compilati non subiscano verifiche all'immissione, ma vengano archiviati a parte. La verifica e l'inserimento nel file definitivo potrà essere effettuato in seguito.
Default: N.

H.2. SEQUENZA DI VERIFICA/CALCOLO? [VERIFY/CALCULATE ORDER:]

È la stessa domanda del D.2.a'., posta in fase di definizione del tipo di derivazione. L'ordine di verifica può interagire con i calcoli e viceversa, per cui la domanda compare anche a questo punto, in fase di definizione del tipo di verifica.
Default: 001.

I. CIFRE DI CONTROLLO? (S/N) [CHECK DIGIT? (Y/N)]

Opzione di utilizzo piuttosto raro. Impone che il numero qui immesso sia divisibile per 11.
Default: N.

J. VERIFICA CAMPO VARIABILITA'? (S/N) [RANGE CHECK? (Y/N)]

Questa opzione consente di imporre che il valore del dato immesso si collochi in un campo di variabilità definito da un minimo e da un massimo.
Default: N.

J.1. VALORE MINIMO DEL CAMPO? [ENTER/CHANGE THE MINIMUM FIELD VALUE:]

Chiede di specificare il valore minimo ammesso per il campo in questione secondo la sequenza ASCII. Es.: per i giorni del mese, minimo 1.
Nessun default.

J.2. VALORE MASSIMO DEL CAMPO? [ENTER/CHANGE THE MAXIMUM FIELD VALUE:]

Analogamente chiede il massimo. Es.: massimo per i giorni 31.
Nessun default.

K. MASCHERA DI IMMISSIONE? (S/N) [EDIT MASK? (Y/N)]

È una caratteristica di grande utilità. Il suo utilizzo permette di effettuare un controllo sulla immissione e sul contenuto di un campo, carattere per carattere. Le numerose opzioni consentono tra l'altro l'inserimento automatico di costanti, la copia del modulo precedente, l'input obbligato in posizioni definite e il controllo su quali caratteri sono ammessi in ogni posizione.
Default: N.

K.1. MASCHERA DI CONTROLLO IMMISSIONE: [ENTRY CONTROL MASK:]

Occorre specificare carattere per carattere il codice di controllo sull'immissione, secondo la tabella visualizzata, per definire l'obbligatorietà o l'opzionalità di immissione, la provenienza del carattere e la presenza o meno di costanti.
Nessun default.

K.2. MASCHERA DI CONTROLLO CONTENUTO: [CONTENENT CONTROL MASK:]

Qui si specifica carattere per carattere il codice di controllo del contenuto secondo la tabella visualizzata. Per definire quali sono i caratteri ammessi in ciascuna posizione.
Nessun default.

L. REGISTRO CARATTERI DI EDITING? [RECORD EDIT CHARACTERS? (Y/N)]

La risposta affermativa a questa domanda fa salvare nel file i caratteri ausiliari del campo, quei caratteri che altrimenti avrebbero solo funzione estetica durante l'immissione: caratteri di riempimento, volanti e costanti definite dalle maschere di controllo. In caso contrario vengono eliminati.
Default: N.

Ricerca per scansione in ordine di file dati

Questo metodo di ricerca consente di scandire il file in ordine sequenziale, visualizzando ogni record. È possibile scandire sia in avanti con ^*N* (Next) che indietro con ^*P* (Preceeding).

Ricerca per scansione in ordine di indice

La scansione per indici ha funzionamento simile alla scansione del file dati, tranne che viene utilizzato il file indice per determinare l'ordine di presentazione. Anche qui valgono ^*N* e ^*P*.

La funzione batch

Esistono talvolta condizioni tali per cui è utile immettere dati in un data base a gruppi di record piuttosto che un record per volta. L'operazione in questo modo è detto batch. Si avvale di un file temporaneo che viene riversato, con le apposite verifiche, sul file principale, in un secondo tempo.
Esempi di utilizzo di batch processing sono:

- Per tenere totali o conteggi sui dati immessi in una seduta.
- Immissione simultanea di dati da parte di più operatori su computer distinti.
- Consente la verifica dei dati in un momento diverso dalla immissione e/o da parte di un diverso operatore, riducendo drasticamente la possibilità di errore.

Varie

Altre operazioni previste da DataStar comprendono:

- Manutenzione del file.
- Aiuti e spiegazioni.
- Uscita dal modulo.

Manutenzione del file

Questa operazione è resa necessaria dalla modalità con cui DataStar gestisce i file. Immettendo un record, DataStar scrive i dati in fondo al file dati, la chiave (A) in fondo al file indice col puntatore al record dei dati corrispondenti e, dopo aver ricercato la chiave (B), precedentemente ad A, e quella seguente (C), segna in corrispondenza di A i puntatori a B e C, per consentire ricerche rapide per bisezionare e scansioni nei due sensi. Per cancellare un record, DataStar semplicemente ne contrassegna l'indice e il record di dati, rendendoli non accessibili, sebbene fisicamente ancora presenti. Per modificare un record DataStar lo marca cancellato e lo riscrive in fondo al file, opportunamente concatenato, come se fosse una normale nuova immissione.
Ne consegue che dopo una fase di immissione l'indice risulta fisicamente non ordinato. DataStar perciò quando ricerca nell'indice (sia per una ricerca che per ottenere i puntatori per l'inserimento di un nuovo record), deve seguire un percorso fisicamente disordinato, lungo la catena dei puntatori. Una ricerca in queste condizioni è necessariamente lunga e laboriosa.
Occorre quindi periodicamente riordinare indice e file dati. Quando infatti l'indice è ordinato, il percorso lungo la catena dei puntatori è regolare, praticamente in sequenza e quindi molto più rapido.
La manutenzione del file (file maintenance) viene effettuato col comando ^*EF*. Altro vantaggio è il recupero dello spazio sul disco risultante dal recupero dello spazio occupato da record marcati "cancellati", ma fisicamente ancora presenti.

Aiuto e spiegazioni

Il comando ^*J*, a livello di mode selector screen (menu principale) consente di avere spiegazioni piuttosto dettagliate sulle opzioni disponibili. Può essere utilizzato come un mini- manuale d'uso entrocontenuto, tipico dei programmi di alto livello.

Uscita dal modulo

Il comando ^*E* consente di uscire da una fase per tornare al menu principale, o uscire completamente dal DataStar.

Il manuale

Come per gli altri prodotti della linea MicroPro, anche il manuale DataStar è piuttosto voluminoso. Di formato grande (26 x 29 cm), in morbido rilegatore ad anelli, comprende oltre 200 pagine di testo, comprendente una sezione dedicata agli esempi di uso ed una per l'installazione. L'organizzazione dei capitoli è abbastanza lineare, peccato tuttavia che manchi un indice analitico.
È compreso col pacchetto una utilissima schedina di riferimento, riassuntiva di tutti i comandi.
L'uso del programma è subordinato alla firma della licenza d'uso. A tal file sono compresi il contratto e la cartolina di registrazione da firmare e restituire alla MicroPro, in modo da essere aggiornato sulle novità e ricevere le nuove versioni.

Conclusioni

È un prodotto che si colloca in una posizione intermedia tra i generatori di maschere video quali QuickScreen (Fox + Geller) e simili, da una parte, e i database veri e propri come dBase II (Ashton-Tate), Selector IV (Micro Ap), Condor dall'altra. Allo stesso tempo, per

Sistema Philips PMDS
Un nuovo sistema che cresce con i vostri programmi di sviluppo.

- Unità a disco rigido
- Potente sistema operativo basato su UNIX
- Possibilità multiuser
- Supporto completo per 8 e 16 Bit
- Estendibilità e upgrading

Il nuovo sistema di sviluppo universale Philips, PM 4422, si affianca sul mercato al ben noto PM 4421 (sistema universale a doppio floppy) con aumentata potenza e flessibilità pur mantenendone compatibilità hardware e software.

• Un investimento sicuro
Il PM 4422 partendo da una configurazione monoutente, può essere espanso, in configurazione multiutente, fino ad un max di 7 stazioni di lavoro.
Così con il crescere dei vostri programmi di sviluppo, il PM 4422 può crescere con essi.

• Memoria di massa su Hard-Disk
La memoria di massa è costituita da un disco rigido Winchester da 5.1/4" inserito direttamente nel sistema, con una capacità di 5 o di 21 Mb. Questa capacità può essere aumentata fino a 147 Mb con l'aggiunta di ulteriori unità a disco.
La memoria interna di sistema può essere espansa da una configurazione standard di 256 Kb ad un max di 1 Mb.

• Sistema operativo basato sullo UNIX*
Il sistema operativo implementato sul PM 4422 è basato sullo UNIX* della Bell di cui son ben note le possibilità multiuser e multitask.

• Supporto 8 e 16 Bit
Il PM 4422 è un sistema di sviluppo universale che supporta microprocessori a 8 e 16 Bit con emulazione in tempo reale.

• Upgrading
I PM 4421 a doppio floppy già esistenti sul mercato possono essere modificati facilmente mediante opportuno kit di conversione in modo da includervi tutti i vantaggi del PM 4422.

* UNIX è un marchio della Bell Laboratories

Philips S.p.A. - Divisione S&I
Strumentazione & Progetti Industriali
Viale Elvezia, 2 - 20052 Monza
Tel. (039) 3635.240/8/9
Telex 333343

Filiali: **Bologna** (051) 493.046
Cagliari (070) 666.740
Padova (049) 632.766
Palermo (091) 527.477
Roma (06) 33.02.344/5/6/7
Torino (011) 21.64.121

PHILIPS

molte applicazioni si colloca in alternativa a generatori di programmi come Quick-n-easi (Standard Micro Systems) e The Last One (AI).
DataStar certamente non si può definire un vero e proprio data base, mentre si colloca a livello nettamente superiore ai generatori di maschere video poichè non solo gestisce l'input su video, ma anche l'immagazzinamento su file e, seppur con qualche limite, la ricerca.

Allo stesso tempo può egregiamente far fronte a molte delle situazioni tipicamente risolvibili con i generatori automatici di programmi di gestione.

Diciamo quindi che DataStar è un ottimo sistema di data entry, che consente una eccezionale flessibilità e facilità di definizione del formato dei dati e un completissimo sistema di verifica della correttezza dei dati immessi. Un livello di verifica di questa potenza è rarissimo nell'ambito del software per microcomputer. Lascia a desiderare, tuttavia, la parte relativa alla gestione dei file. DataStar, infatti, utilizza un metodo di organizzazione dei dati tutto sommato abbastanza rudimentale, che richiede manutenzione periodica per non far degradare rapidamente le prestazioni in fase di scrittura e ricerca.

Si può affermare che DataStar trova il giusto collocamento come utility di front-end, solo per l'immissione dei dati da far elaborare successivamente da programmi appositi. Le sue funzioni di ricerca sono, all'atto pratico, piuttosto elementari e poco potenti. Es.: voglio ricercare tutti i *GIORGIO* nel mio file di amici, in cui sono registrati nel medesimo campo cognome e nome nell'ordine. Come procedo? DataStar non prevede funzioni di ricerca di sottocampi in posizione non definita all'interno di campi interi, e quindi non è possibile ricercare direttamente il dato desiderato. Può supplire a questa necessità un programma apposito oppure SuperSort, utility di ordinamento e selezione di elevatissima qualità, sempre della casa MicroPro.
DataStar è ineguagliato per la grande potenza della sua sezione di immissione, il che dovrebbe condurre a un suo utilizzo come modulo di input in sistemi specifici software. Tuttavia un uso del genere non è di facile implementazione , non essendoci un modo adeguato per concatenare DataStar ad altri moduli di software, rendendo perciò preferibile l'utilizzo di prodotti di minore completezza ma superiore versatilità di implementazione quali Quick-Screen, che oltretutto hanno un costo ben inferiore. Tuttavia non è da trascurare il fatto che un operatore esperto è in grado di "mettere su" un ottimo seppur semplice sistema di bollettazione o fatturazione o simili, con DataStar in 10-15 minuti, pregio di non trascurabile rilievo.
DataStar non contempla la stampa di tabulati diversi dai moduli di immissione, ovvero stampe riassuntive, tabelle con selezioni o calcoli in fase di stampa. Non prevede, cioè, un report generator con capacità di ordinamento, estrazioni e selezione come invece InfoStar, il nuovissimo prodotto MicroPro, compatibile con le maschere DataStar, che rientra a pieno diritto nella classe dei data base management system. Esso riunisce le caratteristiche di DataStar e SuperSort, aggiungendo una sezione di report generator. Ne consegue la capacità di immettere i dati, ordinarli secondo le chiavi desiderate, reperirli, riorganizzarli e stamparli nel formato desiderato, rapidamente e facilmente, generando tabulati di dati successivamente rifinibili con WordStar per aggiungere titoli, note, evidenziazioni e simili.
L'utilizzo di DataStar, in contrapposizione ad altri prodotti, non richiede conoscenza di programmazione, ed è anzi utilizzabile anche da chi è completamente a digiuno di concetti di elaborazione di dati, anche in virtù della completezza del proprio manuale e la struttura tipica dei programmi del sistema MicroPro, in cui l'utente è continuamente guidato e aiutato proprio dal programma stesso.
È quindi doveroso confermare che DataStar risponde pienamente alle aspettative offrendo qualità e affidabilità. ■

Ramtek
Ramtek

Il PET anima il Mini Robot

di M. Esposito e S. Pinto

Si tratta di un programma di impiego generale, implementato su di un CBM 3032 dotato di unità dischi cui è stato collegato un Mini Robot (leggere **Bit** n. 31 per l'esauriente descrizione). Con i suoi codici si ottiene l'apprendimento di movimenti, la loro esecuzione per un numero arbitrario di volte, la registrazione di tali movimenti su disco. Il programma inoltre è dotato di due interessanti subroutine: quella per il ritorno automatico, che realizza il riposizionamento del robottino nelle condizioni in cui si trovava al momento del *RUN* e quella per la "manipolazione dei movimenti", vero e proprio text-editor delle più svariate possibilità di moto.

All'avvio compare sullo schermo il menu principale, mentre la matrice contenente i movimenti da eseguire è vuota. Digitando *A*, il robot si pone nel suo "modo" apprendimento; agendo sulla tastiera secondo le istruzioni apparse sul video si constaterà che il robot si sposta ogni volta nella direzione richiesta. Contemporaneamente i movimenti via via eseguiti vengono registrati nella memoria centrale.

Digitando *E* il programma (e il Mini Robot) vanno nel modo esecuzione, ossia si esegue l'insieme dei movimenti attualmente presenti in memoria, per il numero di volte voluto.

Se si preme *R* i movimenti appresi sono scaricati su disco in un file di dati: il nome di tale file dovrà essere preventivamente inserito alla riga *20050* (se il file viene creato per la prima volta non occorre @).

Premendo *C* i dati sono ripresi dal disco in memoria interna

UN SIMPATICO "BURATTINO" MOSSO DAL PERSONAL COMPUTER

Quando nel numero di Settembre 1982 (pag. 30: "*Arriva Mini Robot il primo personal robot che sbarca in Italia*") ci aspettavamo che questo simpatico personaggio ottenesse un discreto successo per la sua semplicità e versatilità. Ed eccone già una prova dal vivo.

Proviene dal Laboratorio di Controlli Automatici della Facoltà di ingegneria elettronica dell'Università partenopea ed è opera di due laureandi che sottopongono questo software d'uso generale per l'utilizzo del Mini Robot a tutti coloro che possiedono un PET.

Se adesso diciamo che dalla patria di "Pulcinella" non ci aspettavamo niente di meno pertinente i nostri amici napoletani non ce ne vorranno, anzi crediamo si possono dire fieri, assieme ai loro docenti, di aver saputo unire il dilettevole a quell'utile che ogni sana e, quindi, non barbogia didattica dovrebbe sposare.

A tutti gli interessati: buon divertimento con Mini Robot.

REMarks

3	*Viene configurata in uscita la porta parallela del PET (indirizzo 59471).*
4-6	*Sono i dati che, presentati nella successione opportuna alla porta del PET, consentono ai 6 motori del minirobot di muoversi nei due sensi.*
20	*La matrice MN% è quella che contiene tali dati: si è ritenuta sufficiente la possibilità di agire a passo intero, anche se i motori possono essere comandati pure a mezzo passo.*
30-50	*Viene riempita RN%.*
70	*RS% è una variabile il cui valore è fissato a 255 che, presentata sulla porta del PET consente di alzare il bit DI tra un comando al robot e l'altro. Z% contiene il numero di movimenti da eseguire.*
999-1008	*Menu principale.*
1010-1120	*Scelta dell'opzione voluta.*
5000	*CO% è il contatore dei passi eseguiti.*

```
3 POKE(59459),255
4 DATA204,156,60,108,202,154,58,106,200,152,56,104,198,150,54,102
5 DATA196,148,52,100,194,146,50,98,108,60,156,204,106,58,154,202
6 DATA104,56,152,200,102,54,150,198,100,52,148,196,98,50,146,194
7 GOTO20
8 GOTO50251
9 GOTO40020
20 DIMRN%(4,12),MM%(1,100)
30 FORK=1TO12:FORJ=1TO4
50 READRN%(J,K):NEXT:NEXT
70 RS%=255:Z%=0
999 PRINT"J"
1000 PRINT"APPRENDIMENTO:A  ESECUZIONE:E"
1005 PRINT"USCITA:U"
1006 PRINT"REGISTRAZIONE:R  CARICAMENTO:C"
1007 PRINT"MANIPOLAZIONE DEI MOVIMENTI:M"
1008 PRINT"RITORNO AUTOMATICO:Z"
1010 INPUT"COSA VUOI FARE";C$
1100 IFC$="A"THENGOSUB5000
1110 IFC$="E"THENGOSUB10000
1111 IFC$="R"THENGOSUB20000
1112 IFC$="C"THENGOSUB30000
1115 IFC$="U"THENSTOP
1116 IFC$="M"THENGOSUB50000
1117 IFC$="Z"THENGOSUB40000
1120 GOTO999
5000 CO%=0
5100 PRINT"J"
5200 GOSUB61310
5300 PRINT"USCITA:0"
5350 PRINT"PER FERMARE PREMI UN QUALSIASI TASTO"
5400 INPUT"QUALE VUOI MUOVERE";MN%
5800 IFMN%=0THENRETURN
5850 IFMN%<0ORMN%>12GOTO5100
6500 FORI=1TO4
6550 POKE(59471),RN%(I,MN%):POKE(59471),RS%:NEXT
6555 CO%=CO%+1
6560 IFMN%=4THENFORI=1TO4:POKE(59471),RN%(I,11):POKE(59471),RS%:NEXT
6570 IFMN%=10THENFORI=1TO4:POKE(59471),RN%(I,5):POKE(59471),RS%:NEXT
6580 IFMN%=5THENFORI=1TO4:POKE(59471),RN%(I,4):POKE(59471),RS%:NEXT
6590 IFMN%=11THENFORI=1TO4:POKE(59471),RN%(I,10):POKE(59471),RS%:NEXT
6690 GETA$:IFA$<>""THENMM%(0,Z%)=MN%:MM%(1,Z%)=CO%:Z%=Z%+1:GOTO5000
6700 IFZ%=100THENPRINT"HAI ESAURITO I MOVIMEMNTI":RETURN
6800 GOTO6500
10000 IFZ%=0THENPRINT"NON CI SONO MOVIMENTI":FORTT=1TO1000:NEXT:RETURN
10005 INPUT"QUANTE VOLTE";QV%
10010 FORW=1TOQV%
10020 FORY=0TOZ%-1
10100 CB%=0
10150 IFMM%(1,Y)=0THENGOTO10900
10200 FORI=1TO4
10300 POKE(59471),RN%(I,MM%(0,Y)):POKE(59471),RS%:NEXT
10400 CB%=CB%+1
10500 IFMM%(0,Y)=4THENFORI=1TO4:POKE(59471),RN%(I,11):POKE(59471),RS%:NEXT
10600 IFMM%(0,Y)=10THENFORI=1TO4:POKE(59471),RN%(I,5):POKE(59471),RS%:NEXT
10700 IFMM%(0,Y)=5THENFORI=1TO4:POKE(59471),RN%(I,4):POKE(59471),RS%:NEXT
10800 IFMM%(0,Y)=11THENFORI=1TO4:POKE(59471),RN%(I,10):POKE(59471),RS%:NEXT
10900 IFCB%=MM%(1,Y)THENNEXT:NEXT:RETURN
11000 GOTO10200
20000 IFZ%=0THENPRINT"NON C'E' NULLA DA REGISTRARE":FORPK=1TO1000:NEXT:RETURN
20050 OPEN2,8,14,"@0:A$,S,W"
20100 PRINT#2,Z%
20200 FORH=0TOZ%-1
20300 PRINT#2,MM%(0,H)
20350 PRINT#2,MM%(1,H)
20400 NEXT
20600 CLOSE2:RETURN
30000 OPEN3,8,14,"0:A$,S,R"
30100 INPUT#3,Z%
30200 FORH=0TOZ%-1
30300 SW%=0
30310 INPUT#3,M1$
30315 UF=LEN(M1$):U2=UF-1:M2$=MID$(M1$,2)
30330 IFSW%=0THENMM%(0,H)=VAL(LEFT$(M2$,U2)):SW%=1:GOTO30310
30335 MM%(1,H)=VAL(LEFT$(M2$,U2))
30400 NEXT
30450 CLOSE3:RETURN
40000 IFZ%=0THENRETURN
40005 T1%=0:T2%=0:S1%=0:S2%=0:B1%=0:B2%=0:M1%=0:M2%=0:P1%=0:P2%=0:G1%=0:G2%=0
40010 FORHA=0TOZ%-1
40015 ONMM%(0,HA)GOTO40025,40030,40035,40040,40045,40050,40055,40060,9,9,9,9
40020 ONMM%(0,HA)GOTO1,1,1,1,1,1,1,1,40065,40070,40075,40080
40025 T1%=T1%+MM%(1,HA):GOTO40095
40030 S1%=S1%+MM%(1,HA):GOTO40095
40035 B1%=B1%+MM%(1,HA):GOTO40095
40040 M1%=M1%+MM%(1,HA):GOTO40095
40045 P1%=P1%+MM%(1,HA):GOTO40095
40050 G1%=G1%+MM%(1,HA):GOTO40095
40055 T2%=T2%+MM%(1,HA):GOTO40095
40060 S2%=S2%+MM%(1,HA):GOTO40095
40065 B2%=B2%+MM%(1,HA):GOTO40095
40070 M2%=M2%+MM%(1,HA):GOTO40095
40075 P2%=P2%+MM%(1,HA):GOTO40095
40080 G2%=G2%+MM%(1,HA)
40095 NEXT
40099 Z1%=Z%:T3%=T1%-T2%
40100 IFT3%>0THENMM%(0,Z%)=7:MM%(1,Z%)=T3%:Z%=Z%+1:GOTO40200
40130 IFT3%<0THENMM%(0,Z%)=1:MM%(1,Z%)=ABS(T3%):Z%=Z%+1:GOTO40200
40200 T3%=S1%-S2%
40210 IFT3%>0THENMM%(0,Z%)=8:MM%(1,Z%)=T3%:Z%=Z%+1:GOTO40300
40220 IFT3%<0THENMM%(0,Z%)=2:MM%(1,Z%)=ABS(T3%):Z%=Z%+1:GOTO40300
40300 T3%=B1%-B2%
40310 IFT3%>0THENMM%(0,Z%)=9:MM%(1,Z%)=T3%:Z%=Z%+1:GOTO40400
40320 IFT3%<0THENMM%(0,Z%)=3:MM%(1,Z%)=ABS(T3%):Z%=Z%+1:GOTO40400
40400 T3%=M1%-M2%
40410 IFT3%>0THENMM%(0,Z%)=10:MM%(1,Z%)=T3%:Z%=Z%+1:GOTO40500
40420 IFT3%<0THENMM%(0,Z%)=4:MM%(1,Z%)=ABS(T3%):Z%=Z%+1:GOTO40500
40500 T3%=P1%-P2%
40510 IFT3%>0THENMM%(0,Z%)=11:MM%(1,Z%)=T3%:Z%=Z%+1:GOTO40600
40520 IFT3%<0THENMM%(0,Z%)=5:MM%(1,Z%)=ABS(T3%):Z%=Z%+1:GOTO40600
40600 T3%=G1%-G2%
40610 IFT3%>0THENMM%(0,Z%)=12:MM%(1,Z%)=T3%:Z%=Z%+1:GOTO40800
40620 IFT3%<0THENMM%(0,Z%)=6:MM%(1,Z%)=ABS(T3%):Z%=Z%+1
40800 FORY=Z1%TOZ%-1
49150 CB%=0
49200 FORI=1TO4
```

```
49300 POKE(59471),RN%(I,MM%(0,Y)):POKE(59471),RS%:NEXT
49400 CB%=CB%+1
49500 IFMM%(0,Y)=4THENFORI=1TO4:POKE(59471),RN%(I,11):POKE(59471),RS%:NEXT
49600 IFMM%(0,Y)=10THENFORI=1TO4:POKE(59471),RN%(I,5):POKE(59471),RS%:NEXT
49700 IFMM%(0,Y)=5THENFORI=1TO4:POKE(59471),RN%(I,4):POKE(59471),RS%:NEXT
49800 IFMM%(0,Y)=11THENFORI=1TO4:POKE(59471),RN%(I,10):POKE(59471),RS%:NEXT
49900 IFCB%=MM%(1,Y)THENNEXT:RETURN
49950 GOTO49200
50000 IFZ%=0THENPRINT"NON C'E' NIENTE":FL%=1:GOTO50900
50100 A%=0:FL%=0
50200 B%=0:PRINT"J"
50250 ONMM%(0,A%)GOTO50260,50270,50280,50290,50300,50310,50320,50330,8,8,8,8
50251 ONMM%(0,A%)GOTO1,1,1,1,1,1,1,1,50340,50350,50360,50370
50260 S$="TRONCO ANTIORARIO":GOTO50400
50270 S$="SPALLA BASSO     ":GOTO50400
50280 S$="BRACCIO ALTO     ":GOTO50400
50290 S$="MANO BASSO       ":GOTO50400
50300 S$="POLSO ORARIO     ":GOTO50400
50310 S$="PINZA CHIUSA     ":GOTO50400
50320 S$="TRONCO ORARIO    ":GOTO50400
50330 S$="SPALLA ALTO      ":GOTO50400
50340 S$="BRACCIO BASSO    ":GOTO50400
50350 S$="MANO ALTO        ":GOTO50400
50360 S$="POLSO ANTIORARIO ":GOTO50400
50370 S$="PINZA APRE       ":GOTO50400
50400 PRINTA%,S$,MM%(1,A%)
50500 A%=A%+1:B%=B%+1
50550 IFA%=Z%THENFL%=1:PRINT"FINE":GOTO50600
50560 IFB%<12GOTO50250
50600 PRINT"VUOI CORREGGERE?":INPUT"S O N";AA$
50650 IFAA$="N"GOTO50900
50700 IFAA$<>"S"GOTO50600
50750 INPUT"QUALE RIGA";AA%
50800 INPUT"QUANTI PASSI(0 CANCELLA)";MM%(1,AA%):GOTO50600
50900 PRINT"VUOI INSERIRE RIGHE?":INPUT"S O N";BB$
50950 IFBB$="N"GOTO52000
50990 IFBB$<>"S"GOTO50900
51000 INPUT"DOPO QUALE RIGA(SE C'E' NIENTE -1)";DD%
51050 INPUT"QUANTE RIGHE";QQ%
51070 IFZ%+QQ%>100THENPRINT"SONO TROPPE":GOTO51050
51100 IFDD%>=Z%-1GOTO51300
51150 FORL=Z%-1TODD%+1STEP-1
51200 MM%(0,L+QQ%)=MM%(0,L):MM%(1,L+QQ%)=MM%(1,L)
51250 NEXT
51300 PRINT
51310 GOSUB61310
51380 FORKK=DD%+1TOQQ%+DD%
51400 INPUT"QUALE MOTORE";MM%(0,KK):IFMM%(0,KK)<1ORMM%(0,Z%)>12GOTO51400
51500 INPUT"QUANTI PASSI";MM%(1,KK):NEXT
51550 Z%=Z%+QQ%
52000 IFFL%=0GOTO50200
53000 RETURN
61310 PRINT"TRONCO ANTIORARIO:1  TRONCO ORARIO:7"
61320 PRINT"SPALLA BASSO:2   SPALLA ALTO:8"
61330 PRINT"BRACCIO ALTO:3   BRACCIO BASSO:9"
61340 PRINT"MANO BASSO  :4   MANO ALTO:10"
61350 PRINT"POLSO ORARIO:5   POLSO ANTIORARIO:11"
61360 PRINT"PINZA CHIUSA:6   PINZA APERTA:12"
61370 RETURN
READY.
```

5400	*MN% contiene il numero della colonna di RN% in cui ci sono i dati relativi al movimento voluto.*
6500-6555	*Viene eseguito il movimento e incrementato CO%.*
6560-6590	*Vengono realizzati i movimenti di polso e mano, che richiedono l'azionamento, nello stesso verso o in verso opposto, di due motori diversi.*
6690	*Allo stop da tastiera il movimento è registrato in memoria nella matrice MM%: la prima riga indica il tipo di movimento, la seconda il numero di passi (in realtà per passo qui si intende l'insieme di quattro passi del motore).*
6700	*Il limite di 100 non è vincolante, legato com'è unicamente alla seconda dimensione della matrice MM%.*
10005	*QV% contiene il valore del numero di volte per cui il moto globale deve essere ripetuto.*
10100	*CB% conta i passi eseguiti di ogni movimento.*
10200-10800	*Vale quanto detto alle righe 6560-6590.*
20100	*Sul file dati vengono registrati, sequenzialmente, Z% e le prime Z% colonne della matrice MM%.*
30000-30450	*Viene posizionato Z% e viene caricata da disco la matrice MM%.*
40010-40095	*Viene scandagliata la parte non nulla della matrice MM%; al termine i contatori TI%....Q2% contengono il numero complessivo di passi che ogni parte del robot deve eseguire, sia in un senso (i contatori con il pedice 1) che nell'altro (i contatori con il pedice 2).*
40099-40620	*Viene salvato in ZI% il valore di Z%; vengono determinati i passi che ogni motore deve compiere e, se diversi da zero, vengono inseriti nella matrice MM%.*
40800-40950	*Vengono eseguiti tali passi.*
50100	*A% indica il numero della riga che comparirà sullo schermo, eguale al numero di colonna di MM%; FL% è un flag che, posizionato ad 1, indica l'esaurimento della parte non nulla di MM%.*
50200	*B% è il contatore del numero di righe comparse sullo schermo.*
50250-50251	*Il case of serve per l'assegnazione del valore alla stringa S$ a seconda del valore del generico elemento della prima riga di MM%: l'implementazione di tale figura di programmazione, indubbiamente poco ortodossa, è dovuta al fatto che la massima lunghezza di una riga BASIC del PET è di 80 caratteri.*
50400	*Stampa delle informazioni relative al movimento A%.*
50600-50800	*Eventuale modifica del numero di passi di un movimento, fermo restando il motore (si accetta anche il valore 0).*
50900-51100	*Eventuale inserzioni di QQ% nuove righe a partire dalla riga DD%.*
51150-51200	*Vengono spostati in avanti di QQ% posti gli elementi di MM% dal DD% imo in poi.*
51380-51500	*Vengono richiesti i dati relativi ai movimenti da inserire: l'ultimo inserimento è riconosciuto automaticamente.*
51550	*Viene aggiornato Z%.*
61310-61370	*È una subroutine che fà apparire sul video i possibili movimenti del robot e la colonna di RN% che contiene la sequenza di dati ad essi relativi.*

per un'eventuale esecuzione e/o manipolazione. Il nome del file che contiene i movimenti va inserito, preventivamente al *RUN*, alla riga *30000*. Ovviamente il file dati può essere creato in maniera del tutto indipendente dal nostro programma.

La digitazione di *M* conduce alla modalità edit: compaiono così sullo schermo, a gruppi di 12, tutti i movimenti da eseguire con il relativo numero di passi dopo di che, per ciascun gruppo, il programma chiede se si vogliono modificare o cancellare tali movimenti o, eventualmente, inserirne altri.

Quando poi si batte *Z* il robot tornerà, come s'è detto, da solo nello stato iniziale (quello in cui si trovava all'atto del *RUN*) e i movimenti necessari verranno inseriti in memoria in coda agli altri. Questa opzione si è resa necessaria in quanto il nostro Mini Robot non montava gli Switch necessari a chiudere il ciclo di reazione, ossia si è dovuto realizzare, pur nei limiti di ripetitività del robot, un ciclo chiuso via software.

Digitando infine *U* il programma si arresta e i dati presenti in memoria centrale vanno perduti. ■

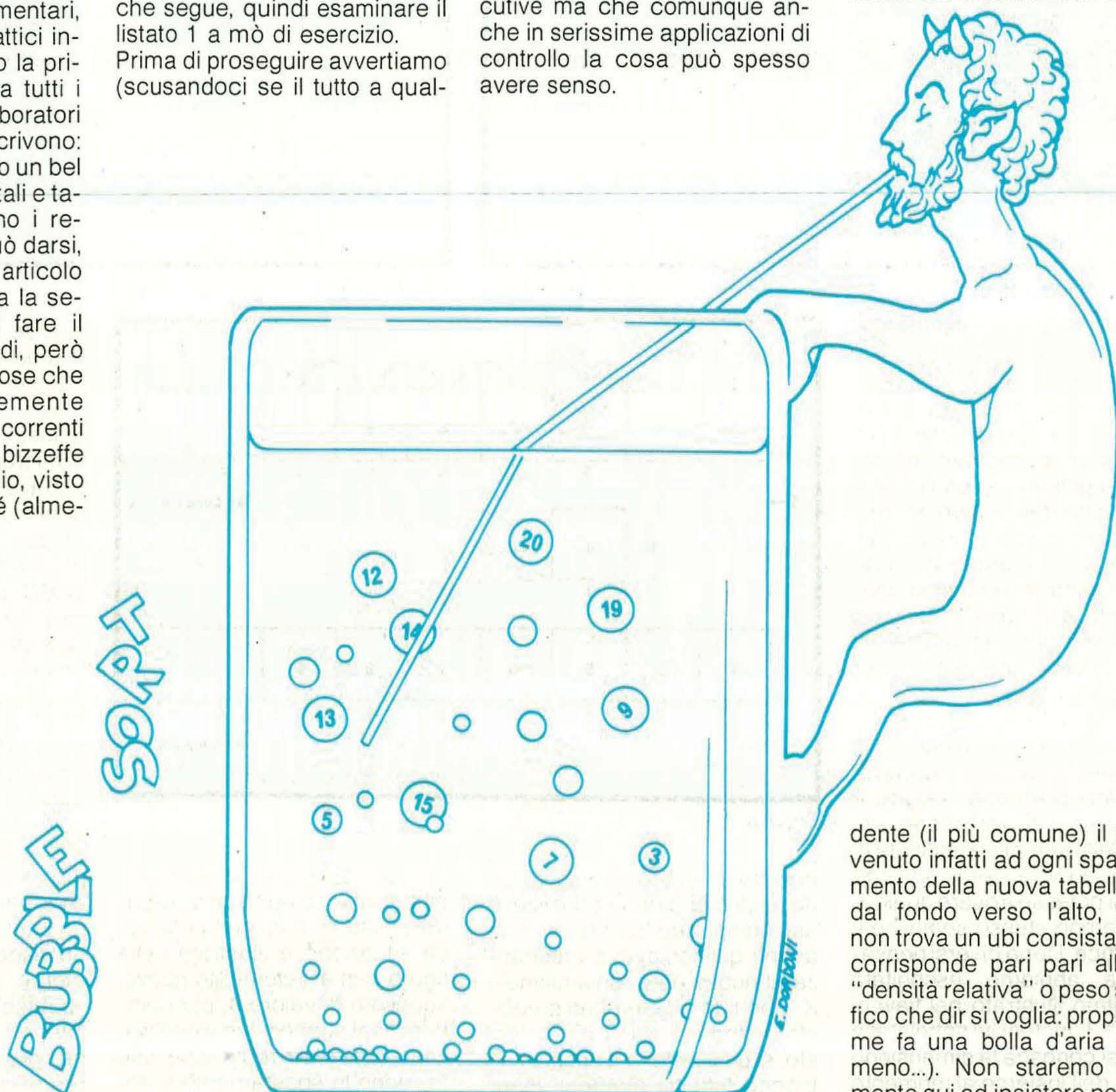
Velocizzare il Bubble Sort

di **G.G.**

Premessa

Nella posta della rubrica Feedback numerose sono le lettere di principianti che, pur apprezzando la "nobiltà" degli argomenti più elevati ed audaci trattati su **Bit**, confessano di stentare a comprenderne molti. La richiesta è allora duplice: da una parte una trattazione in tutti i casi più semplice e divulgativa possibile, dall'altra l'inserimento di questioni più elementari, alla portata di tutti, didattici insomma. Mentre giriamo la prima raccomandazione a tutti i nostri autori e collaboratori (specie coloro che ci scrivono: "Cari redattori, vi mando un bel programmino che fa le tali e talaltre mirabilia. Seguono i remark. Bellino vero?" può darsi, anzi senz'altro: però un articolo è tutt'altra cosa...) circa la seconda cercheremo di fare il possibile in diversi modi, però evitando di affrontare cose che risultano abbondantemente sviscerate sui manuali correnti (ormai ne esistono a bizzeffe anche da noi). O meglio, visto che inventare alcunché (almeno a questo livello introduttivo) è alquanto difficile, ci riferiremo per lo più a varianti di metodi, algoritmi ecc. già noti. O magari arcinoti come appunto il classicissimo "Bubble sort". Una versione di questo, praticamente senza commenti ulteriori, la forniamo nel listato 1. È attuato su PET - CBM Serie 8000 e l'aggettivo "ulteriori" si riferisce al fatto che tale listato ne contiene abbondantemente onde chi non è un novellino integrale se la dovrebbe cavare da sè. In caso contrario il consiglio che diamo è di leggere prima quel che segue, quindi esaminare il listato 1 a mò di esercizio.
Prima di proseguire avvertiamo (scusandoci se il tutto a qualcuno potrà sembrare un poco un pot pourri) che abbiamo fatto un discorso in linguaggio Assembly Z80 e relativo a dati da un byte ciascuno. La cosa va vista perciò in primo luogo come esercitazioncella didattica su una macchina tipo il Nanocomputer, tuttavia la discreta abbondanza di spiegazione dell'algoritmo ne rendono agevole il riciclaggio ovunque. A chi ritenesse assurdo l'ordinamento di un insieme di singoli byte facciamo notare che l'obiettivo scolastico era quello di raffrontare delle velocità esecutive ma che comunque anche in serissime applicazioni di controllo la cosa può spesso avere senso.

Che cosa si vuol fare

Quando si tratta di ordinare un insieme di dati (vettori, tabella ecc.), la più comune tecnica è basata sull'algoritmo del "*bubble sort*". Quando si tratta di aggiungere un nuovo elemento all'insieme, il più corrente criterio consiste nel metterlo in coda a quelli pre-esistenti quindi "sortare" (ma come siamo barbari noi informatici!) il tutto. Il nome di sort a bolle è in questo caso ben giustificato: supponendo un ordinamento ascendente (il più comune) il nuovo venuto infatti ad ogni spazzolamento della nuova tabella sale dal fondo verso l'alto, finchè non trova un ubi consistam che corrisponde pari pari alla sua "densità relativa" o peso specifico che dir si voglia: proprio come fa una bolla d'aria (più o meno...). Non staremo certamente qui ad insistere nella de-

Velocizzare il Bubble Sort

```
89 rem ******************************
90 rem *** per pet-cbm serie 8000 ***
95 rem ***** di giacomo bortone *****
96 rem ******************************
100 rem programma dimostrativo di bubble sort con rappresentazione grafica
110 rem input dati disordinati nel vettore
120 dim k(100):i=0: rem masimo 100 dati da ordinare
130 input "J@Bscrivi i dati disordinati, 0 per finire";n
140 if n = 0 then 155
150 k(i) = n: i = i+1 :goto 130
155 print"J"
160 gosub 1000 :  rem ordinamento
180 print "J":r=i:gosub 500:print:input"Bscrivi un nuovo dato da inserire";k(i)
185 if k(i)= 0 then end : rem termina il programma se si introduce 0
186 i=i+1: rem punta la prina posizione libera nel vettore
190 gosub 2000 : rem ordinamnto di un dato nuovo
200 goto 180
500 rem visualizzazzione
510 print"B"
520 for j = 0 to i-1
525 if j=r then print "B";str$(k(j));" =  ",:goto 540
526 rem  stampa in campo inverso l'ultimo dato mosso, cioe' la bolla
530 print  str$(k(j));"   ",
540 next
550 return
1000 rem ordinamento a bolle
1001 rem l'ordinamento avviene dal basso verso l'alto, i numeri piu' piccoli
1002 rem vengono fatti salire e ogni ciclo controlla una posizione meno
1003 rem il disordine viene confinato in basso
1004 rem nota che per basso si intentono le posizioni di indice piu' alto
1010 r=i: rem r indica l'ultimo dato spostato per la visualizzazione in campo
1011 rem          inverso. e' posto a i, cioe' su un dato che non esiste,
1012 rem          perche' nessun dato e' stato ancora mosso.
1020 gosub 500
1025 if i < 2 then 1090 : rem se c'e' un solo dato nel vettore non si ordina
1027 for x = 1 to i-1: rem ad ogni ciclo controlla una posizione in meno;
1028 rem                ovvero salta quelle gia' ordinate.
1029 fl=0: rem fl resta 0 se non avviene alcuno scambio
1030 for y = i-1 to x step -1: rem spazzola la parte di vettore ancora
1031 rem                           disordinata e muove una bolla verso l'alto
1032 rem                           mettendo in ordine una posizione in piu'
1040 if k(y)>=k(y-1) then 1060
1050 t=k(y):k(y)=k(y-1):k(y-1)=t:fl=1:r=y-1:rem scambia i dati confrontati
1051 rem                                       se necessario, setta fl ed r
1060 gosub 500
1070 next
1080 if fl=0 then x=i-1: rem se non e' avvenuto alcuno scambio forza l'uscita
1081 rem                   dal loop; il vettore e' gia' ordinato.
1085 next
1090 return
2000 rem ordinamento di un nuovo numero introdotto nel vettore gia' ordinato
2010 rem viene eseguito un ciclo solo e' quindi inutile anche fl
2020 rem per altre remark vedi subroutine precedente
2025 gosub 500
2027 if i < 2 then 2090
2030 for y = i-1 to 1 step -1
2040 if k(y)>=k(y-1) then 2080
2050 t=k(y):k(y)=k(y-1):k(y-1)=t:r=y-1
2060 gosub 500
2070 next:goto 2090
2080 y=1:next
2090 return
```

```
LABEL.    CODICE ASSEMBLY          Commenti

          ;SUBROUTINE DI AGGIUNTA/SORT SU VETTORE DI BYTE
          ;INITAB = LABEL INIZIO VETTORE, HL PUNTA A UN BYTE,
          ;DE AL SEGUENTE,BC CONTIENE LUNGHEZZA DEL VETTORE
          ;IL NUOVO ELEMENTO E' SUPPOSTO IN ACCUMULATORE
          ;
UPDSRT    LD HL,INITAB - 1         ;HL punta a sinistra di INITAB
          ADD HL,BC                ;ora HL punta all'ultimo byte
          PUSH BC                  ;salva LUNG sullo stack
          EX AF,AF'                ;salva nuovo elemento
          LD E,L
          LD D,H
          INC DE                   ; DE = HL +1
BCZERO    XOR A                    ;poni zeri nell'Accumulatore
          CP B                     ; B = zero ?
          JR NZ,COMPARA            ;se NO, va alla comparazione
          CP C                     ;anche C è zero?
          JR Z,GIU                 ;se BC = 0 va GIU
COMPARA   EX AF,AF'                ;recupera nuovo elemento
          CP (HL)                  ;cfr. con ultimo(penult. ecc.)
          JR Z,PIUGIU              ;esci dal loop sia se uguali
          JR NC,PIUGIU             ;sia se elemento minore di A
          EX AF,AF'                ;se NO, dopo aver salvato A
          LD A,(HL)                ;poni elem.puntato da HL
          LD (DE),A                ;in quello accanto,puntato da DE
          DEC HL                   ;quindi decrementa HL
          DEC DE                   ;DE nonché
          DEC BC                   ;il de-conta cicli
          JR BCZERO                ;e salta ove si vede se BC=0
GIU       EX AF,AF'
          LD (DE),A                ;metti nuovo venuto al suo posto
          POP BC                   ;ripesca LUNG
          INC BC                   ;aumentala di un'unità
          RET                      ;e rientra al MAIN
```

scrizione dell'algoritmo gassoso, noto ormai al punto di esser venuto a noia ai più. Ricordiamo solamente che in esso è in gioco un numero di scambi tra elementi proporzionato, nel caso dell'aggiunta, ai posti che il nuovo dato deve salire per trovare la sua giusta collocazione. Poiché uno scambio prevede *tre* spostamenti e cioè: lo spostamento di un dato in un'area di comodo; lo spostamento del secondo dato nel primo e, per terza cosa, il movimento del comodo nell'area del secondo dato, ad una anche generica riflessione il metodo non ci appariva particolarmente veloce e comunque intuivamo che, nel caso della semplice aggiunta di un dato ad un insieme *già ordinato*, si potesse trovare qualcosa al tempo stesso semplice e più veloce. Dopo diversi pensaripensa abbiamo escogitato l'algoritmo illustrato nel flaw di figura 2. L'ipotesi ivi sottointesa è che si conosca la dimensione *LUNG* dell'insieme al momento

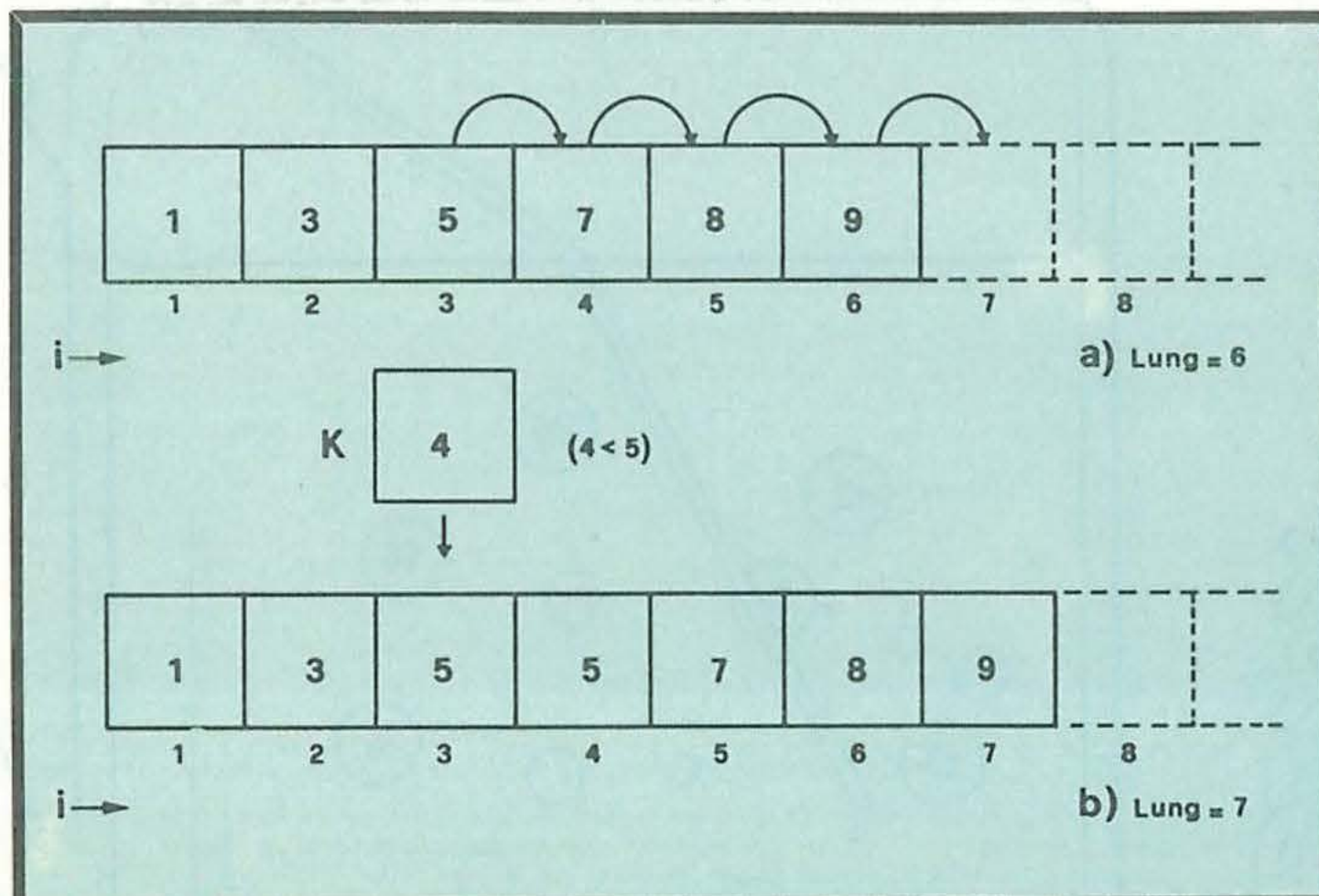



Figura 1.

di entrare nella nostra subroutine. L'idea originaria consisteva nel comparare (a ritroso o in avanti, qui non aveva importanza) il nuovo dato, chiamiamolo *K*, con tutti gli elementi preesistenti. Individuato il posto giusto, si provvedeva a spostare in blocco tutti gli elementi maggiori del nostro che veniva così, alfine, inserito nel "buco" creatosi: tutto di sua competenza. La situazione è illustrata nella figura 1 in a: l'elemento nuovo, supposto di valore 4, per comparazioni successive viene trovato *minore* del terzo (che vale 5); dopo lo spostamento a destra del blocco dei dati 5, 7, 8 e 9 si ha la situazione illustrata in b e a questo punto basta ficcare 4 al posto del "vecchio 5". Questo metodo, codificato per il microprocessore Z80, comportava però un calcolo preliminare, necessario alla determinazione della lunghezza del blocco da spostare (e da porre in *BC*, per sfruttare la celebre istruzione *LDIR*, qui impiegata facendo puntare *HL* e *DE* sul fondo) sulla base della vecchia lunghezza dell'intero insieme e della posizione trovata per il nuovo dato. In linguaggio macchina tutto ciò non è difficile ma tedioso e relativamente "time consuming". Alla prova di velocità si verificava perciò quanto previsto, ossia: per tabelle sufficientemente lunghe (diciamo un centinaio di byte) si otteneva un risparmio di tempo mediamente del 50% sul classico "bubble", mentre per lunghezze inferiori il peso dell'elaborazione preliminare appena detta rendeva la cosa meno conve-

niente.
In seguito ad una discussione con altra gente si è escogitato qualcosa che dovrebbe andar meglio. Per non farla troppo lunga lasciamo parlare direttamente il diagrammino a blocchi. Detta, al solito, *LUNG* la dimensione attuale dell'insieme e ipotizzando che essa possa essere (inizialmente) nulla - il che significa vettore vuoto - si va dopo aver posto l'indice i al valore *LUNG*, a vedere se *LUNG* è per caso di valore zero. Se questo avviene, il nuovo arrivato *K* è posto al.... posto dell'*EL* (i+1), quindi si aumenta *LUNG* di un'unità. Il che poi, come ovvio, in questa evenienza pone *K* nel primo ed unico elemento dell'insieme ora non più vuoto e di dimensione, ora, *LUNG* = *1*. Se invece la tabella ha almeno un elemento si compara *EL* (i), ossia l'ultimo, poi il penultimo ecc., con *K*. Finchè il primo risulta *maggiore* di *K* si provvede a spostare *EL (i)* nell'*EL (i+1)*, poi decrementando l'indice i.

A proposito: per semplicità si è supposto che lo spazio disponibile per la crescita del nostro insieme sia praticamente infinito. Ai perfezionisti il non difficile compito di aggiungere un test di limite. Proseguendo la comparazione tra elementi del vettore e *K*, "a ritroso" si arriva o alla condizione in cui *K* è maggiore o uguale oppure *q* quella in cui l'indice i diventa nullo. In entrambi i casi si perviene al blocco in cui l'elemento nuovo *K* è posto *accanto* a quello appena oggetto di comparazione, si accresce *LUNG* di uno e si rientra dalla subroutine. Il lettore potrà rendersi conto che l'algoritmo funziona in tutti i casi: ovviamente va bene quando si esce dl loop senza alcun giro, cioè con *i=LUNG*; in una situazione intermedia, ad ogni passaggio per il blocco segnato con *X* sulla sinistra, si crea, gradualmente, un buco - meglio un doppione - come già illustrato nella figura 1 b. Quanto al caso in cui si perviene al valore i=0, il blocco finale (segnato con *Y* sulla destra) pone il nuovo dato, che evidentemente è il minore di tutti, al primo posto. Si noti che sono evitati rischi di comparazione con dati al di fuori di quelli pre-esistenti. In conclusione, il confronto viene effettuato partendo dal basso e, grazie a ciò, non occorrono calcoli particolari. Si fa notare che anche ora si ha la tentazione di fare un'analogia fluidica. Onde evitare confusioni nominalistiche potremmo denominare il tutto sort "*per lacune*" (chi conosce i semiconduttori drogati di tipo P intende). Una possibile implementazione in BASIC potrebbe essere la seguente:

```
200 FOR I = LUNG TO 1 STEP - 1
210 IF LUNG = O OR EL (I) <=K GOTO 230
220 EL (I+1) = EL (I) : NEXT I
230 EL (I+1) = K : LUNG = LUNG + 1 : RETURN
```

Figura 2 - Aggiunta di un elemento ad una tabella ordinata, senza necessità degli scambi richiesti dal classico "bubble sort".

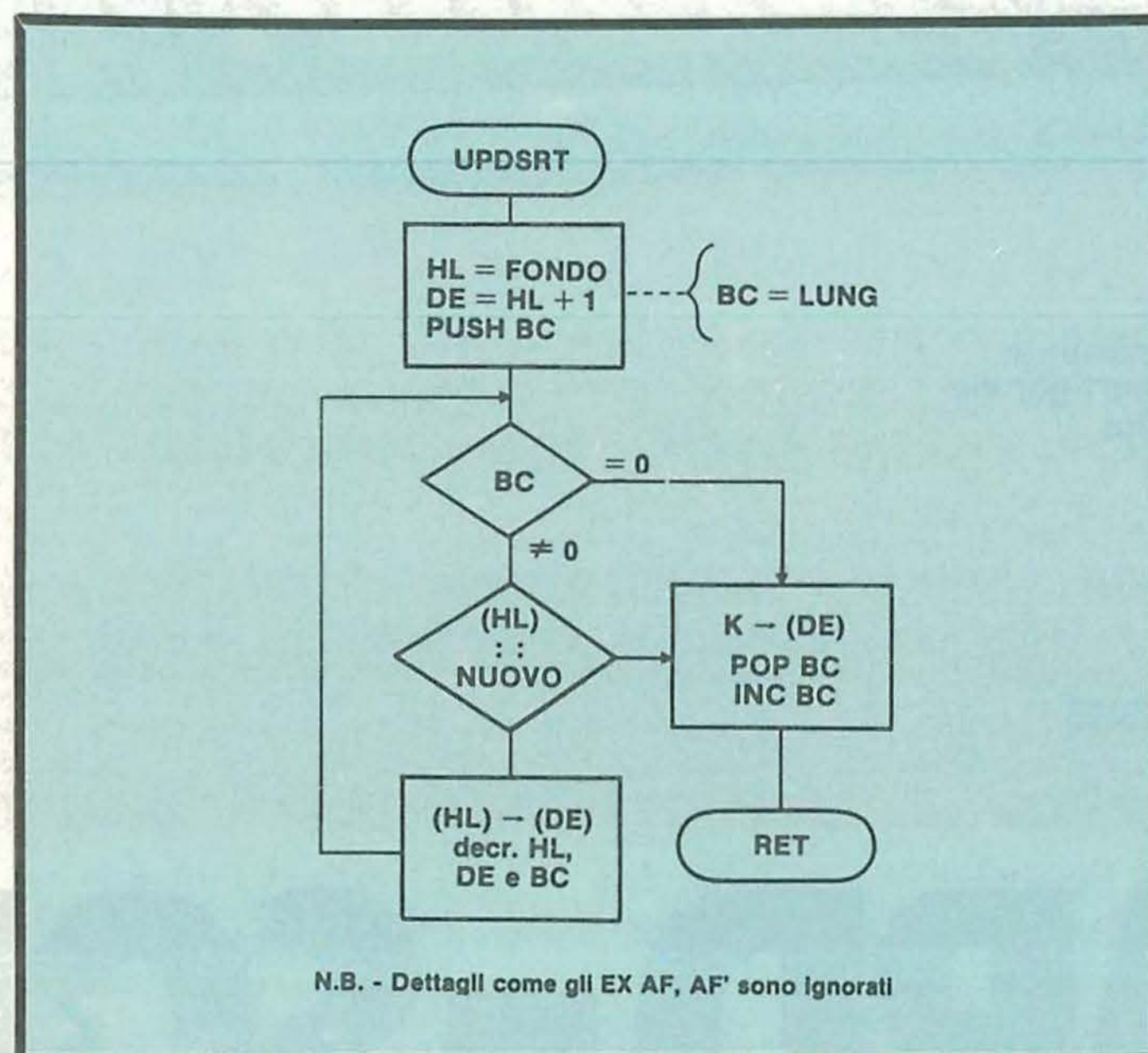



Figura 3 - Esecuzione (su un vettore di byte) per eventuali prove di velocità in linguaggio macchina Z80.

Si ricorda, per chi avesse dubbi in proposito, che da una *FOR-...NEXT I* con *STEP -1*, all'uscita da *NEXT I* il valore di I è proprio zero, il che va a fagiolo al caso nostro.

Codifica Assembly Z-80

Riportata con tutti i suoi bravi commenti in cima e a fianco, non dovrebbe presentare difficoltà. Diciamo solo alcune cose. Per poter giocare con elementi che, in BASIC, individuavamo tranquillamente con *I* e *I+1* abbiamo pensanto di usare le coppie di registri-puntatori *HL* e, rispettivamente, *DE* (quest'ultimo fatto inizialmente pari ad *HL + 1*, mentre *HL = INITAB - 1 + BC* punta, com'è giusto, all'ultimo elemento. Il nuovo, sitto nell'accumulatore, viene salvato mediante la (veloce) *EX AF, AF'*. Si ha poi la comparazione della *LUNG* contenuta in *BC* con zero, comparando sia *B* che *C* con *A* (preliminarmente azzerato con l'*XOR*) con se stessa. Una spiegazione merita il salto a *PIUGIU* sulle condizioni *NC* e *Z* (ossia "minore-o-uguale": si è supposto di operare con dati positivi nel range 0 ÷ 255). La cosa dipende dai salvataggi-recuperi di *A* tramite gli scambi con il registro alternativo *A'*: proveniendo dalla condizione *BC = 0* occorre saltare alla label *GIU* ove si fa *EX AF, AF'* per il repêchage, mentre proveniendo dall'interno del loop questo è già stato effettuato (alla label Compara), onde la *EX AF, AF'* va by-passata. Si noti anche il *PUSH BC* in cima ed il *POP BC* in fondo per il salvataggio-recupero della *LUNG*, dato che qui *BC* viene utilizzato come deconta-cicli. Chi ben riflette noterà che un flow che più da vicino illustra la soluzione Assembly Z-80 è del tipo riportato in figura 3.
Si è fatto un discorso di vettori di byte singoli sia per semplicità sia perché l'obiettivo era una prova di velocità (ben riuscita, si assicura). Chi vuol fare di meglio si può accomodare: dovrà sostituire i confronti e gli spostamenti tra singoli byte con adeguate routine relative a voci e campi. Buon lavoro.

■

Le nuove frontiere del calcolo strutturale

di G. Forcolini

Dato un solido di forma e dimensioni qualsiasi, sollecitato da forze esterne ed interne, la sua risposta elastica non è sempre ricavabile con i metodi del calcolo infinitesimale. Operando invece una *discretizzazione*, cioè immaginando il solido suddiviso in *elementi* interconnessi in punti discreti, tale risposta è ottenibile per via numerica, attraverso la risoluzione di un sistema di equazioni lineari. Si usa allora la matematica finita nelle formulazioni matriciali, per la quale sappiamo che è particolarmente adatto l'elaboratore elettronico.

Calcolo infinitesimale e calcolo numerico

Per secoli, l'ambizione dei matematici è stata quella di leggere i fenomeni naturali attraverso una rete di proposizioni matematiche. La scienza delle costruzioni - da Galileo fino al nostro secolo - si è sviluppata sulla base di geniali dimostrazioni matematiche che, prendendo le mosse dell'analisi infinitesimale del continuo, giungevano a rappresentare il comportamento elastico in termini di equazioni differenziali.
Dall'inizio di queste ricerche trascorrerà lungo tempo prima della fattiva utilizzazione dei principi statici nella pratica progettuale e costruttiva. Non è senza significato che gran parte degli algoritmi del calcolo infinitesimale siano opera di grandi matematici. Proprio questa origine strettamente disciplinare ha reso spesso difficile l'innesto nella pratica costruttiva. Infatti, quando la struttura acquista una certa complessità, la soluzione analitica ricavata per il caso più semplice e generale, non può che essere approssimata. E provare ad affrontare nuovamente il problema matematico sulla base della sua particolare configurazione porta spesso ad onerosi quanto funambolici stratagemmi di calcolo, per ammettere alla fine l'irrisolvibilità del problema.
Si è resa perciò necessaria una maggiore aderenza tra strumentazione matematica e pratica progettuale. È quello che si è verificato nel corso di questo secolo con l'elaborazione del metodo cosiddetto "agli elementi finiti". Precisamente nella seconda metà degli anni sessanta, ingegneri e matematici insieme hanno cominciato a riconsiderare l'intera problematica, puntando innanzitutto sulla ricerca di nuovi e più flessibili linguaggi analitici. Non a caso ciò ha coinciso con le prime importanti realizzazioni nel campo dei computer; ed in parallelo si affacciava sulla scena della ricerca interdisciplinare l'analisi numerica, sottodominio della matematica volta alle applicazioni pratiche e - in questo ambito - alla formulazione di algoritmi destinati specificatamente all'elaborazione elettronica.

Gli "elementi finiti"

Non vogliamo dilungarci sulla "filosofia" del metodo. In poche parole, il solido è ridotto ad elementi di forma identica e di grandezza opportuna. Di tali elementi vengono preventivamente analizzate le caratteristiche di reazione elastica a spostamenti o a forze applicate nei punti nodali di interconnessione. Messi in archivio questi dati numerici sotto forma di matrici, si costruisce il modello matematico che riproduce il comportamento elastico del solido trasformato dalla *discretizzazione* in ideale struttura piana o spaziale. Tale modello, come abbiamo detto, non è altro che un sistema di equazioni lineari ottenuto imponendo determinate condizioni (di *equilibrio* e di *congruenza*) all'insieme degli elementi finiti. Esso esprime le forze note agenti sulla struttura in funzione degli spostamento incogniti (o viceversa), attraverso noti coefficienti di rigidezza o di deformabilità.
Ottenuti tali spostamenti si tornerà agli elementi e si calcoleranno le sollecitazioni nei loro punti nodali. È comodo scrivere il sistema in forma matriciale ed istruire il computer sui procedimenti più rapidi per arrivare all'unica soluzione possibile e reale.
La fase più delicata - come si comprenderà - consiste nella riduzione in elementi, nella scelta della loro forma e dimensione, e nella numerazione da dare agli spostamenti incogniti al fine di ottenere una matrice di composizione tale da consentire di risparmiare quanta più memoria e di velocizzare al massimo l'iter di calcolo. Ed è anche la fase in cui interviene l'esperienza del progettista e la sua *sensibilità statica*. È altresì evidente che la fedeltà del modello matematico al comportamento elastico della struttura dipende dal numero degli elementi di suddivisione e della loro dislocazione nelle zone in cui v'è da attendersi un maggior tasso di variazione del regime delle sollecitazioni.

Occupazione di memoria e tempi d'esecuzione

Il BASIC in versione Microsoft, adottato dalla maggioranza dei personal, non contiene istruzioni per eseguire operazioni matriciali (trasposizione, inversione di una matrice, somma, sottrazione e prodotto tra più matrici). Se vogliamo, in fondo si tratta di una limitazione solo apparente. Si è preferito non implementare routine standardizzate, ma lasciare all'operatore il compito di scrivere sottoprogrammi opportunamente studiati per il tipo di matrice che egli si trova a manipolare. Del resto anche nei BASIC estesi, contenenti istruzioni apposite, viene solitamente fissato il tetto massimo di 2.500 locazioni (dimensioni 50 x 50). In Microsoft BASIC l'istruzione *DIM* è tutto quello che serve, naturalmente corredata da una buona dose di loop nidificati.
L'impegno maggiore è naturalmente quello di ridurre al minimo l'occupazione della memoria operativa e i tempi di esecuzione. Il fine si persegue in molti modi: innanzitutto facendo uso esteso della programmazione ottimizzata e strutturata (**Bit** se n'è occupato nei numeri 23/1981 e 24/1982), per la quale però sappiamo che il BASIC, in tutte le sue versioni, non è molto adatto. Altro accorgimento largamente usato è quello di archiviare nella memoria di massa del *mini floppy* le matrici degli elementi con i valori assegnati attraverso i dati di *input*, per poi richiamarle singolarmente durante l'esecuzione del programma. Ulteriore attenzione va posta nell'uso di algoritmi che, nel ricavare risultati, prevedano la loro collocazione in celle di memoria già occupate da valori numerici

che, non intervenendo più nel calcolo, possono essere in tal modo cancellati. Di solito nella matrice finale molte locazioni non contengono altro che zeri, pertanto nel calcolo conviene ignorarle ed inoltre ridurre la stessa matrice ad un vettore, ciò che consente, pur restando pienamente all'interno del metodo, un notevolissimo risparmio di memoria.
Questa matrice che si potrebbe definire di simulazione del comportamento fisico della struttura, si presenta simmetrica, a diagonale dominante e definita positiva. Per il suo trattamento conviene usare algoritmi particolari. Nel testo ne proponiamo alcuni di base, in forma elementare. Solitamente formulati in Fortran, in Algol o in PL/1, particolare cura è stata qui usata nel tradurli in Microsoft BASIC.
Ci si sorprenderà a constatare come un impiego oculato degli algoritmi dell'analisi numerica conduca a risparmi di memoria veramente considerevoli. Se a questo aggiungiamo il ricorso metodico alla programmazione in una parola *razionalizzata*, cioè a tutte le tecniche di scrittura che contribuiscono a velocizzare i programmi e a renderli facilmente leggibili, possiamo concludere che il personal è un valido strumento anche nel caso di strutture complesse. Soprattutto in questo campo applicativo, dove i calcoli raggiungono spesso un volume notevole e i dati in gioco sono numerosi, è di capitale importanza programmare in modo chiaro e leggibile in modo da rendere sempre trasparente il cammino logico percorso. Come è facile intuire, i risultati numerici di per sè non sempre riescono a garantire l'efficienza del programma.

Algoritmi di algebra lineare in BASIC

Trasposizione di matrice. Data una matrice quadrata di ordine *n*, la sua trasposta è ricavata scambiando ogni riga con la corrispondente colonna. Ad esempio:

$$A=\begin{bmatrix}3 & 2 & 5\\ 6 & 1 & 9\\ 4 & 7 & 2\end{bmatrix}$$

$$A^T=\begin{bmatrix}3 & 6 & 4\\ 2 & 1 & 7\\ 5 & 9 & 2\end{bmatrix}$$

Il seguente algoritmo trasforma *A(I,J)* in *(A^T(I,J)*:

```
10  FOR I=0 TO N-1
20  FOR J=I+1 TO N
30  S=A (I,J) : A(I,J)=A(J,I) : A(J,I)=S
40  NEXT J,I
```

Prodotto di A(M,N) e B(N,R) in C(M,R). Data *A* di ordine (*m* x *n*) e *B* (*n* x *r*), il seguente algoritmo ottiene il loro prodotto in una terza matrice *C* di ordine (*m* x *r*):

```
10  FOR I=0 TO M
20  FOR J=0 TO R
30  FOR K=0 TO N
40  C(I,J)=C(I,J) + A(I,K) ★ B(K,J)
50  NEXT K,J,I
```

Sistemi di equazioni lineari.

Matrice dei coefficienti delle incognite triangolare superiore. Ad esempio un sistema del tipo:

$$\begin{aligned}2x + 3y + 4z &= 20\\ 0 + 2y - z &= 3\\ 0 + 0 + 7z &= 21\end{aligned}$$

si risolve componendo i coefficienti delle incognite nella matrice *A (2,2)* e i termini noti (20,3,21) nel vettore *B(2)*. L'algoritmo proposto usa il vettore *B* per collocarvi i valori delle incognite x,y,z:

```
10  FOR I=N TO 0 STEP -1
20  B(I) = B(I) / A(I,I)
30  IF I=N THEN 60
40  FOR J=I + 1 TO N
50  B(I) = B(I) - A(I,J) ★ B(J)/A(I,I) : NEXT
60  NEXT
```

Triangolare inferiore. Sistema del tipo:

$$\begin{aligned}2x + 0 + 0 &= 2\\ 3x + 2y + 0 &= 7\\ 4x - y + 7z &= 23\end{aligned}$$

(la matrice dei coefficienti è la trasposta di quella dell'esempio precedente) sono a matrice triangolare inferiore. Il relativo algoritmo di soluzione è:

```
10  FOR I=0 TO N
20  B(I) = B(I) / A(I,I)
30  FOR J=0 TO I-1
40  B(I) = B(I) - (I>0) ★ A(J,I) ★ B(J) / A(I,I)
50  NEXT J.I
```

L'operatore relazionale (I>0) è necessario perchè il BASIC non ignora il contenuto del loop se il valore da incrementare supera il valore che segue il *TO*.

Algoritmo di Cholesky. I sistemi di equazioni lineari che si incontrano frequentemente in svariati campi applicativi, hanno la matrice dei coefficienti simmetrica e definita positiva. In questi soli casi si può procedere con la sua *fattorizzazione* e la conseguente risoluzione di due sistemi triangolari. Il teorema di Cholesky dimostra che *A* (simmetrica e definita positiva) è uguale al prodotto di due matrici triangolari L^T e *L* di cui la prima (triangolare inferiore) è la trasposta della seconda (triangolare superiore). Perciò: $A = L^T \star L$ per cui: $Ax = B \rightarrow L^T \star L \star x = B \rightarrow L^T \star y = B$ ed infine: $L \star x = y$. Dalla soluzione dell'ultimo sistema si ottengono le incognite stoccate nel vettore disponibile *B* già contenente i termini y. Il triangolo superiore di *A* viene utilizzato per i valori di *L*. L'algoritmo che segue è la base per successive ottimizzazioni, secondo la particolare configurazione della matrice *A*. ■

```
10  FOR I=0 TO N
20  FOR J=I TO N : S=0
30  FOR K=0 TO I - 1 : S =(I>0) ★ (S + A(K,I) A(K,J)) : NEXT
40  IF J=I THEN 60
50  A(I,J) = (A(I,J) -S) / A(I,I) : GOTO 70
60  A(I,J) = SQR (A (I,J) - S)
70  NEXT J,I
80  FOR I=0 TO N
90  B(I) = B(I) / A(I,I)
100 FOR J=0 TO I - 1
110 B(I) = B(I) - (I>0) ★ A(J,I) ★ B(J) / A(I,I)
120 NEXT J,I

130 FOR I=N TO 0 STEP - 1
140 B(I) = B(I) / A(I,I)
150 IF I=N THEN 180
160 FOR J= I + 1 TO N
170 B(I) = B(I) - A(I,J) ★ B(J) / A(I,I) : NEXT
180 NEXT
```

Harden Italia. Il salto di qualità.

Dal personal computer al professional computer.

Nel quadro di una filosofia aziendale in evoluzione, Harden Italia riconferma la validità della proposta del Sirius 1. Il Sirius 1, con tutta la potenza del suo microprocessore a 16 bit, con 5 MHz, e una memoria centrale che può arrivare 896 KBytes, è uno dei più avanzati della nuova generazione dei Personal.

Oltre ad una enorme capacità di archiviazione dei dati (dai 1240 KBytes del Sirius 1 agli 11.840 KBytes del Sirius 1b) il Sirius può contare su alcune caratteristiche che un tecnico e un professionista non possono non apprezzare: dall'interfacciamento con due porte seriali e una parallela programmabile da software, ai sistemi operativi (MS-DOS della Microsoft e CP/M86 della Digital Research), fino ai linguaggi di alto livello come il BASIC-86 (interprete e compilatore), l'Assembler, il COBOL, il Fortran, il Pascal.

Oltre che sul software vero e proprio (programmi come il Dbase II, il SuperCalc, il Multiplan o l'Harden-text e l'Harden-data) il Sirius 1 si avvale dei così detti "Tool Kits", una serie cioè di utilities compatibili con qualsiasi linguaggio che permettono una stesura dei programmi più facile e più completa come ad esempio l'AutoSort, il FABS, una gestione sofisticata IS, ecc. In più, il Sirius 1 è distribuito e assistito dalla Harden Italia su tutto il territorio nazionale.

Per saperne di più sul Sirius 1, sui suoi programmi o su dove sono i punti di vendita Harden più vicini, chiamare (0372)-63136 oppure (02)-651645: risponde la Harden Italia.

sirius

Harden Italia S.p.A. Direzione generale e uffici commerciali
20121 Milano - via dei Giardini, 4 - tel. (02) 651645
Sede operativa e uffici commerciali
26048 Sospiro (CR) - tel. (0372) 63136 - telex: 3205881

SIRIUS 1 CONFIGURAZIONE BASE
(128 KBYTES RAM, 1240 KBYTES FLOPPY DIS
DA OGGI L.6.500.000

è in edicola il nuovo numero

- VAI IN USA COL CONTE DRACULA
- I NUOVI EXTRA JOY STICK
- È NATO UN ANDROBOT
- CREATIVISION ALL'ATTACCO
- GARE! SUPERGARE!! SUPERVIDEOGARE!!!

UNA PUBBLICAZIONE DEL GRUPPO EDITORIALE JACKSON

Fisilab: il personal computer nel laboratorio di fisica

a cura di **A. Cavalcoli**

Parte seconda

PRESENTAZIONE

In questo numero di **Bit Education** presentiamo, la seconda, e ultima, parte del progetto *Fisilab*.
Dopo la descrizione del software relativo ai segnali di *START* e *STOP*, viene ora descritta la sezione operativa delle due esperienze di cinematica. Infine, viene dato uno spazio particolare a tutta una serie di considerazioni didattiche.

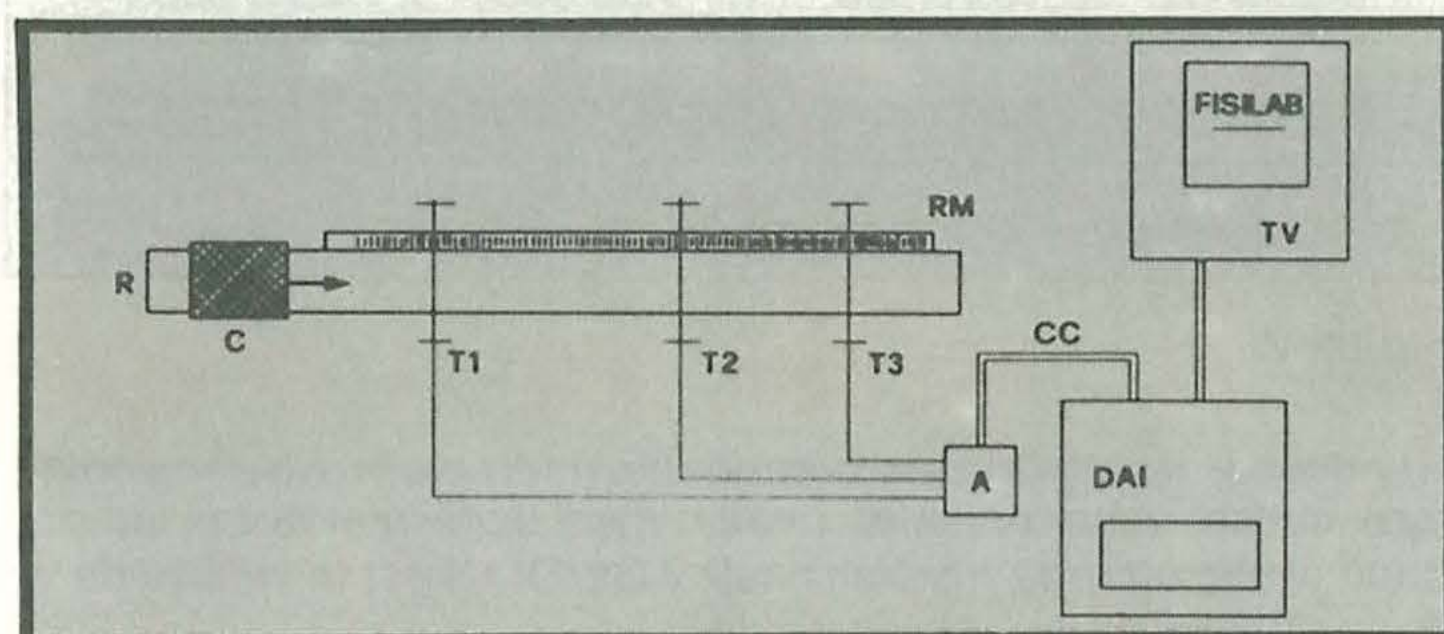
Dopo aver descritto la parte del programma relativa ai segnali di *START* e di *STOP* e al disegno delle tabelle dei dati, in questo secondo articolo vediamo la parte operativa vera e propria che gestisce le due esperienze di cinematica: il moto rettilineo uniforme ed uniformemente accelerato.
Si presuppone la presenza in laboratorio di una rotaia a cuscino d'aria (apparecchio ormai disponibile in tutti gli istituti scientifici) su cui possono scorrere senza attrito uno o più carrelli capaci di far scattare un relay con segnali ottici (o magnetici) in unione ad un dispositivo amplificatore.
Normalmente il relay comanda a sua volta la partenza di un cronometro elettromeccanico od elettronico.
Nel nostro caso invece la chiusura dei contatti del relay servirà a far partire il cronometro interno al computer come indicato nella prima parte (**Bit** n. 37) e nella figura 1.

R = Rotaia a cuscino d'aria
C = Carrello
T1, T2, T3 = Traguardi ottici (o magnetici)
A = Amplificatore
RM = Righello millimetrato per prelevare direttamente la posizione dei vari traguardi
CC = Cavetto di collegamento (descritto nella prima parte)

Figura 1 - Mostra di un possibile cablaggio dei vari componenti.

E' tassativamente necessario che, dovendo usare più traguardi ottici (o magnetici), l'amplificatore comandi un solo relay per cui i vari dispositivi dovranno essere collegati in parallelo.
Poichè i segnali di *START* possono essere prelevati in due modi diversi, per rendere più lineare il commento del programma, li descriveremo separatamente con due esempi diversi, il primo relativo al moto rettilineo uniforme, il secondo al moto uniformemente accelerato.
Sperando che fin qui sia stato capito tutto, facciamo *RUN* e leggiamo i primi messaggi che compaiono sul video:

STUDIO SPERIMENTALE DELLA CINEMATICA E DELLA DINAMICA

1 - Moto rettilineo uniforme
2 - Moto rettilineo uniformemente accelerato
3 - Secondo principio della dinamica
4 - Esperienza di Chadwick

Premere il numero corrispondente all'esperienza desiderata

Naturalmente decidiamo per il n. 1 dopo di che abbiamo nuove istruzioni con le richieste dei dati (figura 2).
(Per distinguere le richieste del calcolatore dalle nostre risposte, indicheremo queste ultime in grassetto).

MOTO RETTILINEO UNIFORME

Verifichiamo che: (X2 - X1)/(t2 - t1) = costante

Gli spazi sono misurati in: **cm**
La posizione iniziale è: Xo = **40** cm
Con quanti traguardi? (1 - 2 - 3): **2** (*)
1° traguardo a **67** cm
2° traguardo a **100** cm

Con quante prove? (1 - 2 - 3 - 4 - 5): **1**(*)

(Nota: le risposte con (*) non vengono stampate).

Figura 2.

Subito dopo la nostra ultima risposta si forma la seguente tabella (figura 3).

Spazi in cm →	Ds 1 = 27.0	Ds 2 = 33.0
1ª Prova Dt		

Figura 3.

Prima di continuare dobbiamo fare alcuni commenti:

1° Osservando i dati Ds 1 e Ds 2 stampati in alto possiamo notare che Ds 1 è la distanza tra il primo traguardo e la posizione iniziale Xo e che Ds 2 è la distanza tra il 2° ed il 1° traguardo.

2° Non avendo un dispositivo adatto ad imprimere al carrello C sempre la stessa velocità occorre indicare *una sola prova* in quanto più prove non avrebbero nessun legame tra loro; non preoccupiamoci per ora, questa limitazione sarà superata con indicazioni successive.

3° Sul percorso del carrello *devono essere stati posti 3 traguardi* di cui il primo T1 serve per lo *START* del cronometro (questo è dunque il primo modo), il secondo T2 per il primo intervallo di tempo, il terzo T3 per il secondo intervallo di tempo e contemporaneamente per lo *STOP* (vedere figura 1); lo spazio prima di T1 servirà per imprimere un impulso al carrello C.

4° Evidentemente, quando il carrello attraversa il 2° traguardo, il cronometro deve azzerarsi e poi ripartire; questo avviene per mezzo della variabile di controllo *Z%* (linea *16160*) che in questo caso è = 0 (linea *1060*).

Facciamo partire il carrello e prendiamo per la prima volta i tempi: i risultati sono indicati nella figura 4.

Spazi in cm →	Ds 1 = 27.0	Ds 2 = 33.0
1ª Prova Dt	1.82 sec	2.26 sec
Ds/Dt =	14.83 cm/s	14.6 cm/s
Premere<C> per continuare, <F> per finire		

Figura 4.

Dunque, dopo aver scritto i due valori di Dt, il computer stampa sotto ogni colonna i rapporti Ds/Dt per la verifica della loro uguaglianza.

A questo punto, il commento da fare e che inizia con la famosa frase: *"NEI LIMITI DEGLI ERRORI SPERIMENTALI......"* crediamo non rientri negli scopi di questo articolo!

Come notiamo però dalla figura 4 il programma dà la possibilità di continuare.

Premendo <C> infatti si sostituirà la scritta:

STESSI DATI? (S/N)

Naturalmente risponderemo *S* per riportarci alla situazione della figura 3 (vedi il precedente commento n. 2) e potremo fare quante verifiche vogliamo; rispondendo *N* ci riporteremo invece alla situazione della figura 10.

Si premerà invece *F* per finire l'esperienza.

Vediamo adesso la 2° esperienza premendo il n. 2 dopo il *RUN*. Il nuovo messaggio e le nuove richieste diventano quelle di figura 5.

MOTO RETTILINEO UNIFORMEMENTE ACCELERATO

Considereremo sempre Vo = 0 e to = 0 e quindi studieremo l'equazione oraria: x(t) = 1/2 a t ↑ 2 + Xo

Verifichiamo allora che: 2 (x(t) - Xo)/t ↑ 2 = costante

Gli spazi sono misurati in: **cm**
La posizione iniziale è: Xo = **0** cm
Con quanti traguardi? (1 - 2 - 3): **3** (*)
1° traguardo a **10** cm
2° traguardo a **40** cm
3° traguardo a **90** cm

Con quante prove? (1 - 2 - 3 - 4 - 5): **5**(*)

Figura 5.

Dopo l'ultima nostra risposta si forma la tabella della figura 6.

Spazi in cm →	Ds 1 = 10.0	Ds 2 = 40.0	Ds 3 = 90.0
1ª Prova Dt			

Figura 6.

Anche per questo esempio sono necessari alcuni commenti:

1° I tre valori di Ds sono tutti riferiti alla posizione iniziale Xo = 0.

2° Le condizioni che generano un moto uniformemente accelerato sono facilmente riproducibili per cui è possibile ripetere più prove (5, nel nostro esempio).

3° Poichè la velocità iniziale deve essere 0, il segnale di *START* deve coincidere con la partenza del carrello C, quindi lo stesso dispositivo che lo lascia partire deve comandare anche la prima chiusura del relay (e questo è il secondo modo); i traguardi T1 e T2 serviranno allora per il 1° e 2° intervallo di spazio, mentre T3 per il terzo spazio e per lo *STOP*.

4° Poichè i Ds sono riferiti sempre a Xo, il cronometro *non deve* azzerarsi ad ogni passaggio dai traguardi; a questo ci pensa ancora la variabile *Z%* posizionata adesso ad 1 (linea *2060*).

Facciamo partire il carrello e ripetiamo 5 prove: tempi e risultati sono riportati nella figura 7.

Spazi in cm →	Ds 1 = 10.0	Ds 2 = 40.0	Ds 3 = 90.0
1ª prova Dt	0.73 sec	1.46 sec	2.2 sec
2ª prova Dt	0.77 sec	1.58 sec	2.31 sec
3ª prova Dt	0.79 sec	1.62 sec	2.38 sec
4ª prova Dt	0.76 sec	1.49 sec	2.28 sec
5ª prova Dt	0.75 sec	1.5 sec	2.28 sec
Media Dt	0.76 sec	1.53 sec	2.29 sec
2 Ds/Dt ↑ 2 =	34.62 cm/s↑2	34.47 cm/s↑2	34.32 cm/s ↑2
Premere <C> per continuare, <F> per finire			

Figura 7.

Commenti: rispetto al caso precedente (una sola prova), il calcolatore, questa volta, prima dà i valori medi degli intervalli di tempo, dopo di che stampa i risultati finali: 2 Ds/Dt ↑ 2 per la verifica della loro uguaglianza ("... nei limiti ...!!).

Naturalmente alla richiesta di fondo pagina video rispondiamo *F*!

Coloro che hanno seguito il primo articolo del programma *Fisilab* dovrebbero aver capito come vengono immessi i dati e come vengono successivamente elaborati.

In quest'ultima parte allora, descrivendo le due esperienze che rimangono, dedicheremo un po' di spazio ad alcune considerazioni didattiche limitando all'indispensabile il commento alle varie linee del listato.

A questo punto quindi il "dialogo" è aperto: tutti coloro che vorranno pronunciarsi con suggerimenti o con osservazioni possono farlo scrivendo direttamente all'autore; se risulteranno interessanti, ci faremo cura di ritornare su questo argomento con un articolo

apposito nel quale riassumeremo tutte le "varie voci" dei colleghi che avranno scritto.
Incominciamo dunque dalla 3° esperienza: "verifica del 2° Principio della Dinamica".
Quando viene affrontato questo argomento con una classe, gli alunni non conoscono ancora - con rigore scientifico - i concetti di massa, forza e peso, per cui il programma *Fisilab* è strutturato in modo da sfruttare questa loro "ignoranza" per lasciare all'insegnante il compito di "tirare le somme" con la dovuta precisione tecnica.
In altre parole questa terza parte del programma serve a prendere ed elaborare i dati senza fare fisica perchè questo compito dovrà spettare all'insegnante partendo dai risultati finali.
Una disposizione ideale dei vari apparecchi è indicata nella figura 8 dove il carrello C, tenuto inizialmente fermo nella posizione di partenza, è tirato da un "pesino P" di 2 grammi che può cadere liberamente in verticale. L'unico traguardo T1 servirà per lo *STOP* in quanto il segnale di *START* deve essere comandato dallo stesso dispositivo che lascia partire il carrello.

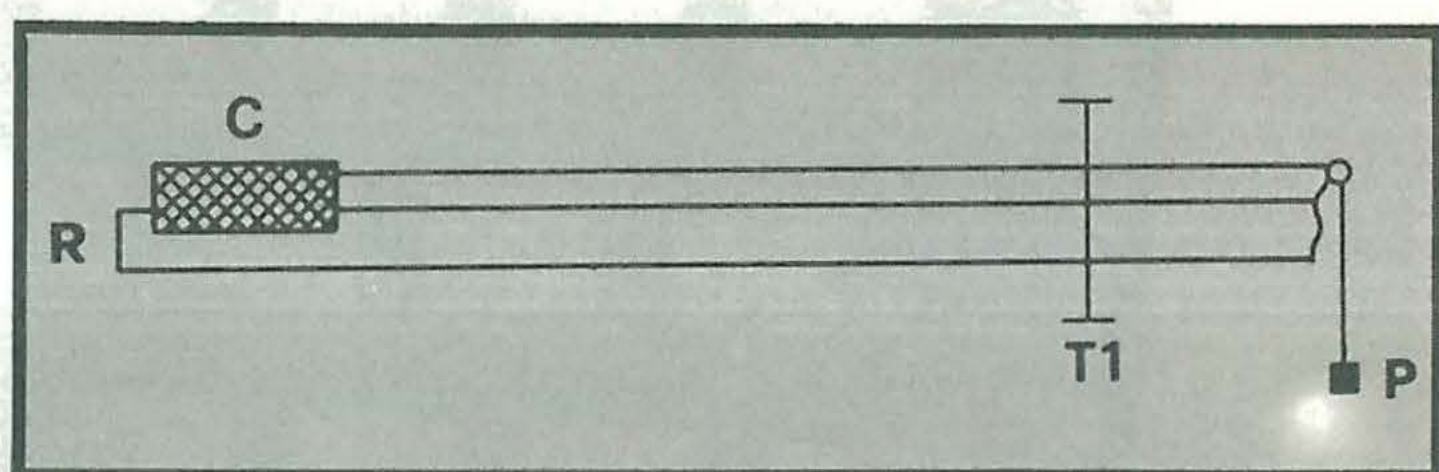



Figura 8.

Dopo aver risposto col n. 3 alla richiesta iniziale, le indicazioni che compaiono sul video (figura 9) servono a far conoscere il criterio sperimentale seguito (linee *3030 - 3095*).

SECONDO PRINCIPIO DELLA DINAMICA	
Verifichiamo la relazione che c'è tra forza applicata ad un corpo ed accelerazione acquistata dal corpo stesso. Verifichiamo cioè che FORZA/ACCELERAZIONE = costante	
1° prova: una sola forza	- un solo carrello
2° prova: due forze	- un solo carrello
3° prova: una sola forza	- due carrelli
4° prova: due forze	- due carrelli
Premere un tasto per scrivere i dati	

Figura 9.

Per gli studenti questo schema di lavoro risulta molto chiaro: hanno a disposizione un "peso" di 2 o 4 grammi che tira uno o due carrelli, c'è dunque "una forza" che tira un carrello oppure "due forze" che tirano un carrello ecc. ecc.
"Per loro" la logica dell'esperienza è evidente e l'insegnante può fare a meno di usare inizialmente la parola "massa".
(*N.B.* Per questo motivo il rapporto F/a sarà lasciato senza unità di misura).
Andiamo avanti.
La richiesta dei dati è simile ai casi precedenti (vedere figura 2) con in più, nella prima riga di pagina video, il messaggio: (linee *3115 - 3120*).
(Alla richiesta del numero dei traguardi, rispondere 1) e poi sotto:

FORZA TRAENTE (in g) =

Considerando la massa del carrello di almeno 100 g, si deve rispondere con un numero piccolo (ad es. 2) a tale domanda, così l'errore "teorico" che si commette considerando in moto "solo" la massa del carrello è trascurabile.
Al numero di prove, possiamo rispondere con 5.
Una grossa novità comparirà quando il calcolatore darà i risultati (linee *3170 - 3440*) in fondo alla tabella (che noi riportiamo però solo per l'ultima prova nella figura 10), i calcoli vengono eseguiti considerando:

- la forza traente = 2 ± 0,1 g.
- l'intervallo di tempo = 13,85 ± 0,01 secondi
- lo spazio Ds = 90 ± 0,1 cm.

quindi il valore dell'accelerazione, ad esempio, è ottenuto, dopo aver trovato a(max) ed a(min), con l'espressione:

$$a = (a(max) + a(min))/2 \pm (a(max) - a(min))/2$$

È superfluo dire quale utilità offra il PC in questo caso: i risultati - media Dt - accelerazione - F/a - sono stampati in poco più di un secondo!
Eseguite le 4 prove nella successione chiaramente indicata sul lato destro della tabella (figura 10), i 4 risultati del rapporto F/a vengono riportati in una nuova tabella simile alla seguente (figura 11): (linee *3510 - 3700*).

Spazi in cm →	Ds 1 = 90.0	
1ª prova Dt	13.8 sec	4° ESPERIENZA
2ª prova Dt	13.88 sec	2 FORZE
3ª prova Dt	13.86 sec	2 CARRELLI
4ª prova Dt	13.86 sec	
5ª prova Dt	13.85 sec	
Media Dt	13.85 sec	
accelerazione = 0.93 ± 0.0 cm/sec ↑ 2		
F/a = 4/26 ± 0.11		
Premere un tasto per continuare		

Figura 10.

Due spunti per la discussione finale:
a) si può notare (leggendo verticalmente) che il rapporto F/a risulta costante (... nei limiti ...) quando non cambia il numero dei carrelli trainati;
b) si può notare (leggendo orizzontalmente) il rapporto F/a raddoppia col raddoppiare del numero dei carrelli.
Crediamo che la strada per arrivare all'equazione del 2° Principio della Dinamica e alla definizione di massa sia aperta e crediamo che a questo punto il nostro compito, per questa esperienza, sia finito.
Ultima tappa: "esperienza di Chadwick".
Com'è noto, lo scienziato inglese Chadwick riuscì a misurare la massa del neutrone considerando gli urti di questa particella contro protoni della paraffina e protoni dell'azoto applicando le leggi di conservazione dell'energia e della quantità di moto.
E' molto facile simulare questa esperienza in laboratorio e quindi è consigliabile proporla ai nostri alunni tanto più che, dal punto di vista didattico, è una valida applicazione dei principi di conservazione sopra detti che rimangono, il più delle volte, solo accennati.
Le indicazioni sulla procedura da seguire si hanno con il 1° messaggio sul video (linee *4020 - 4110*).

sinclair

I PUNTI DI FORZA

- Grafica a 256x192 punti-schermo.
- 8 colori indipendenti per testo, sfondo, riquadro.
- Comandi di suono modulabili in frequenza e durata.
- Vera tastiera multifunzione con maiuscole e minuscole. Tutti i tasti con funzione di ripetizione.
- Compatibile con teletext.
- Alta velocità LOAD e SAVE: 16k byte/ 100 sec.
- Funzioni VERIFY e MERGE per programmi e archivi.
- BASIC Sinclair esteso con funzioni a 1 tasto; controllo di sintassi.
- Ampio software su cassetta.
- Perfettamente compatibile con la stampante ZX.
- Due modelli:
  16k byte ROM e 16k byte RAM,
  16k byte ROM e 48k byte RAM.

CPU E MEMORIA ESPANDIBILE

Microprocessore Z80A.
ROM 16k contenente l'interprete BASIC e il sistema operativo.
RAM 16k espandibile a 48k byte.

TASTIERA MULTIFUNZIONE

È dotata di 40 tasti mobili che danno accesso a caratteri maiuscoli e minuscoli ASCII.
Tutte le parole chiave del BASIC sono ottenibili tramite un singolo tasto. Inoltre sono disponibili 16 caratteri grafici, 22 codici di controlli colore e 21 caratteri grafici definibili dall'utente.
Tutti i tasti sono dotati di ripetizione automatica.
Sono presenti i comandi di cursore.

GRAFICA AD ALTA RISOLUZIONE

Lo **ZX Spectrum** può essere collegato direttamente a qualsiasi televisore a colori PAL o in bianconero.
Sono generati 8 colori: nero, blu, rosso, magenta, verde, azzurro, giallo, bianco - sui televisori in bianconero essi appaiono come una regolare scala di grigi.
La grafica è a 256x192 punti. I testi sono visualizzati in 24 linee di 32 caratteri ciascuna. Testo e grafica possono essere sovrapposti. Le istruzioni grafiche BASIC permettono il tracciamento di punti, linee, cerchi ed archi di cerchio.

Di ogni carattere viene memorizzato il colore, il colore dello sfondo, lo stato fisso o lampeggiante, la luminosità normale o extra, il modo diretto o inverso.

Gli attributi di ciascun carattere possono essere determinati indipendentemente da quelli dei caratteri presenti contemporaneamente sullo schermo.
Normalmente le prime 22 righe visualizzano il listato mentre le ultime due sono riservate per evidenziare la linea di programma in fase di editing.
Per l'editing si ricorre ai comandi di cursore.

SUONO

L'altoparlante interno può riprodurre una scala di più di 10 ottave, esattamente 130 semitoni, attraverso il comando BASIC BEEP. Le prese di tipo jack nella parte posteriore del computer permettono la connessione con altoparlanti e amplificatori esterni.

OPERAZIONI E FUNZIONI

Oltre ai normali operatori matematici sono presenti funzioni trascendenti: seno, coseno, tangente e inverse; logaritmi naturali ed esponenziali, funzione segno, valore assoluto, integer, radice quadrata; pigreco; generatore di numeri casuali.
I numeri memorizzati occupano 5 byte: il campo è da 3×10^{-39} a 7×10^{38} con accuratezza di 9½ cifre decimali.
Si possono trattare numeri binari, effettuare operazioni logiche, definire funzioni da parte dell'utente.
È presente un meccanismo completo di DATA, che include i comandi READ, DATA e RESTORE.
Si possono effettuare operazioni sulle stringhe: concatenazione, segmentazione, estrazione di parti.
I vettori possono essere multidimensionali con indici che partono da 1.

IL BASIC E LA GESTIONE DEI FILE

Il libro si rivolge in modo particolare a chi già conosce il Basic e desidera poter realizzare programmi che prevedano l'uso di file residenti su disco. Dopo aver preso in esame, utilizzando numerosi esempi pratici, le particolarità del Microsoft, si passa alla descrizione delle istruzioni necessarie ad una corretta gestione dei file su disco, sia ad accesso diretto che sequenziale. Una terza parte del libro è infine interamente dedicata alla esposizione dei metodi pratici per l'uso dei file ad accesso diretto e dei data base.

Cod. 515H **L. 11.000 Pagg. 164**

50 ESERCIZI IN BASIC

Una raccolta completa e progressiva di esercizi riguardanti matematica, gestione, ricerca operativa, gioco e statistica. Ciascun esercizio proposto comporta l'enunciazione e l'analisi del problema, la risoluzione mediante flow-chart e commenti, così come un programma che implementa la soluzione, illustrato da semplici esempi rappresentativi.
Questo metodo mette in grado il lettore di verificare passo passo le sue conoscenze e il livello di apprendimento raggiunto.

Cod. 521A **L. 13.000 Pagg. 208**

75 PROGRAMMI IN BASIC PER IL VOSTRO COMPUTER

Il volume raccoglie ben 75 programmi originali scritti in un Basic generico, utilizzabili, salvo poche e semplici modifiche, sulla maggior parte dei personal computer in commercio, a cassetta come a floppy disk. Per ciascuno, dopo una descrizione introduttiva, viene fornito il listing e un campione di esecuzione. Così come sono, i programmi proposti (tutti verificati), costituiscono un valido ausilio per chiunque debba risolvere problemi di matematica, statistica, finanza o, genericamente, di pratica utilità.

Cod. 551D **L. 12.000 Pagg. 196**

GIOCARE IN BASIC

Nei giochi, il lettore può ritrovare tutte quelle situazioni reali di programmazione che gli saranno indispensabili nella comprensione e realizzazione di qualsiasi applicazione interattiva del proprio computer, anche le più sofisticate. Questo senza annoiarsi, ma entrando da subito all'interno della materia per imparare a comprendere il BASIC, il proprio computer e i computer in genere.

Cod. 522A **L. 20.000 Pagg. 324**

DIVISIONE LIBRI

CEDOLA DI COMMISSIONE LIBRARIA

Ritagliare (o fotocopiare) e inviare a Gruppo Editoriale Jackson Via Rosellini, 12 - 20124 Milano

Nome e Cognome ____________________

____

Indirizzo ____________________

____

Cap. ________ Città ____________________

Provincia ____________________

Codice Fiscale (indispensabile per le aziende)

Inviatemi i seguenti libri:

Codice Libro	Quantità	Codice Libro	Quantità	Codice Libro	Quantità	Codice Libro	Quantità

☐ Pagherò al postino il prezzo indicato + l. 2.000 per contributo fisso spese di spedizione

☐ Allego assegno n° .... di L. ....

Data .... Firma ....

	1 CARRELLO	2 CARRELLI
1 F.	F/a = 2.08 ± 0.1	F/a = 4.35 ± 0.22
2 F.	F/a = 2.18 ± 0.06	F/a = 4.26 ± 0.11

Figura 11.

ESPERIENZA DI CHADWICK CON LA SEGUENTE SIMULAZIONE

Il carrello A di massa (incognita)	Ma = il protone
Il carrello B di massa (nota)	Mb = protone paraffina
Il carrello C di massa (nota)	Mc = protone azoto

(N.B. Inizialmente i carrelli B e C devono essere fermi).

Misureremo V'b = velocità di B dopo l'urto con A
Misureremo V'c = velocità di C dopo l'urto con A

Determineremo la massa di A (Ma) con la formula

$$Ma = \frac{V'c - Mc - V'b - Mb}{V'b - V'c}$$

(Premere un tasto per continuare)

Figura 12.

Una disposizione dei vari pezzi può essere quella di figura 13:

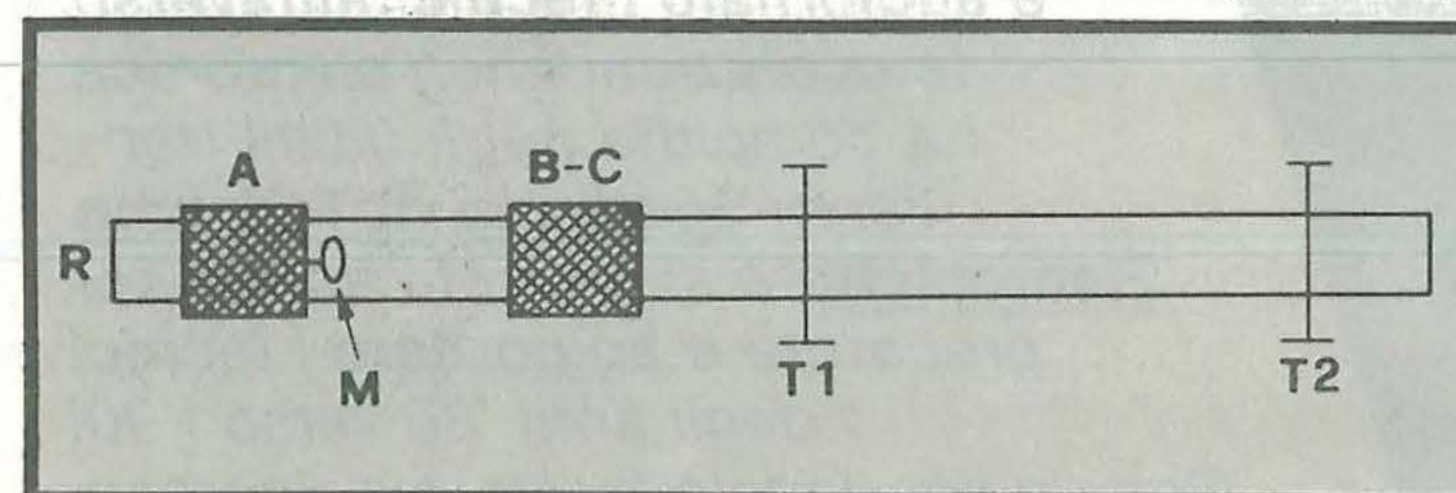



Figura 13.

(Alla richiesta del numero dei traguardi, rispondere 1)

MASSA (in g) DEL CARRELLO B Mb = **150** g
MASSA (in g) DEL CARRELLO C Mc = **200** g

Gli spazi sono misurati in **cm**
La posizione iniziale è: Xo = **45** cm
Con quanti traguardi (1-2-3)? **1**(*)

1° traguardo a **110** cm

Con quante prove (1-2-3-4-5)? **5**(*)

Figura 14.

dove abbiamo indicato con M una molla per rendere "elastico" l'urto tra A e gli altri due carrelli.
I consigli di carattere tecnico sono:
a) i due traguardi T1 e T2 servono rispettivamente per lo *START* e per lo *STOP* per cui dobbiamo rispondere con 1 alla richiesta relativa;
b) per evitare che il carrello A oltrepassi il primo traguardo T1 dopo l'urto con B o con C, dobbiamo scegliere le masse di B e C non minori di quelle di A;
c) questa volta è necessario possedere un dispositivo che imprima al carrello A sempre la stessa velocità perchè la formula che ci dà Ma deriva dal presupposto che Va (velocita di A prima dell'urto con le due masse Mb ed Mc) sia sempre la stessa.
La pagina video dei dati si presenta come in figura 14 (linee *4160 - 4210*).
Nelle figure 15a e 15b sono riportati i dati di una prova effettuata come visibile in figura 15a, 15b.
Dopo di che, l'ultima pagina video diventa come in figura 16: (linee *4520 - 4700*)
A questo punto, per la discussione finale, basta porre su una bilancia il carrello A, misurarne la massa e fare il confronto col risultato dato dal calcolatore.
E che DAI ce la mandi buona!

Spazi in cm →	Ds 1 = 65.0	
1ª prova Dt 2ª prova Dt 3ª prova Dt 4ª prova Dt 5ª prova Dt	2.7 sec 2.75 sec 2.68 sec 2.69 sec 2.73 sec	**1° ESPERIENZA** **Urto tra A e B**
Media Dt	2.71 sec	
V'b = 23.99 ± 0.13 cm/sec		
(Premere un tasto per continuare)		

Figura 15a.

Spazi in cm →	Ds 1 = 65.0	
1° prova Dt 2° prova Dt 3° prova Dt 4° prova Dt 5° prova Dt	3.3 sec 3.21 sec 3.24 sec 3.24 sec 3.26 sec	**1° ESPERIENZA** **Urto tra A e C**
Media Dt	3.25 sec	
V'c = 20.0 ± 0.09 cm/sec		
Premere un tasto per continuare		

Figura 15b.

ECCO I DATI ED I RISULTATI OTTENUTI

Mb = 150 ± 1 g   Mc = 200 ± 1 g

V'b = 23.99 ± 0.13   V'c = 20.0 ± 0.09 cm/s

da cui ricaviamo la misura di Ma:

$$Ma = \frac{V'c - Mc - V'b - Mb}{V'b - V'c} \; 100.13 \pm 14.86 \text{ g}$$

Premere C per continuare, F per finire

Figura 16.

La biblioteca di Bit

a cura di **A. Cavalcoli**

Il primo libro della rubrica di questo mese è una pubblicazione inglese, dedicata ai criteri di scelta di un piccolo sistema gestionale.
Anche se la mia critica non è molto favorevole, ho ritenuto utile proporlo all'attenzione del lettore per sottolineare lo stato dell'arte di questo tipo di pubblicistica.
Il secondo è un manuale SGS, non certo fresco di stampa, ma non superato dai tempi.
Spesso ci si dimentica dell'esistenza di queste pubblicazioni, che invece è utile tenere d'occhio.

Choosing and using a business microcomputer
R. Bradbeer, J. Allason, B. Miles, R. Webb
Ed. Gower

Testo in inglese che, come dice il titolo, si pone l'obiettivo di aiutare quanti si pongano il problema della scelta di un microcomputer per applicazioni gestionali.
Spesso questi libri di "criteri di scelta", non mantengono quanto promesso dal titolo, ed anche questo purtroppo non si discosta dalla media.
Gli autori cercano di individuare alcune norme di comportamento per il potenziale utente, fornendo definizioni e chiarificazioni, per lo più molto vaghe, o, comunque, troppo ampie per poter essere prese con sicura base culturale.
Forse più interessante la sezione dedicata ai "case studies", vale a dire allo studio di recente nascita, che dovrebbe servire di monito e di esempio per i futuri acquirenti di piccoli sistemi.
La realtà proposta è però inglese, quindi poco trasferibile alle aziende italiane, pur essendo almeno europea.
Alla fine, il libro riveste una sua utilità, che però è molto ridotta, rispetto alle giuste aspettative del lettore.
E questa non è tanto una nota di demerito per gli autori, ma, ancora una volta, la verifica della quasi completa carenza di buona letteratura sull'argomento scelte e valutazioni di piccoli sistemi.

Z80 CPU - Set di istruzioni

Nel recensire libri e letteratura tecnica in genere, non bisogna mai scordarsi della produzione dei costruttori, che spesso offre interessante, se non fondamentale, materia di studio e consultazione.
Spesso il lettore si rivolge al libro di testo, quando invece dovrebbe valutare la presenza di manuali specializzati nel catalogo di quanti producono il computer, il sistema, oggetto del suo studio. È il caso della SGS, che nella sua manualistica propone questo Z80, Set di istruzioni.
Le varie istruzioni del microprocessore Z80 sono suddivise nei classici gruppi, all'interno dei quali ogni singola istruzione è dettagliatamente e completamente descritta, secondo il seguente schema:

- l'operazione svolta dall'istruzione;
- il format dell'istruzione, suddiviso in codice operativo e con la notazione binaria completa;
- una descrizione dettagliata dell'operatività, che non dovrebbe lasciare dubbi sulla funzione svolta, con un elenco di tutte le possibili combinazioni;
- il dettaglio dei cicli macchina, degli stati T e la durata, sulla base di un clock di 4 MHz, da definirsi medio, se si considera che le attuali versioni sono: 2.5, 4, 6, ed anche 8 MHz, con il recente Z80H;
- infine, l'elenco e lo studio degli eventuali flag coinvolti dalla istruzione.

Quindi, più che una lettura, un testo di consultazione, come tanti altri indispensabile per gli operatori del settore. ■

Il ricettario

Questo mese il menu è più disorganico che mai, come c'era da aspettarsi. Speriamo che faccia la gioia degli amanti del pot-pourri. Comunque ce n'è per tutti i gusti, anche se difficilmente elencabili in colonne separate, come avviene nei ristoranti cinesi.
Vi sono ricettine piccine piccine per Sinclairisti o per tutti-o-quasi (tra cui si segnala un rimedio che restituisce alla virgola la sua funzione di segno d'interpunzione anche nella registrazione dei file); c'è l'apporto del nostro beneamato dottor Staderini Stranamore (lo chiamo così perchè forse apprezza più l'anatomia di Apple II che quella del corpo umano ...) che generosamente, insieme ai suoi trucchi, svela o rinfresca diverse cosucce su questa ancor inesplorata disciplina.
Ad esse abbiamo aggiunto un dulcis-in-fundo sotto forma di una rubrica nella rubrica: "La sfida del mese". Può darsi che in questa prima veste essa sia imperfetta, ma l'importante è, come al solito, l'aver gettato un sasso. In piccionaia o nello stagno, fate voi. Adesso voglio vedere se non ci si muove più alacremente, senza esagerare, però: già ho la scrivania rigurgitante di contributi volonterosi, ma alquanto debolucci. Sotto dunque con le idee (anche di hardware, dimenticavo!).

Da numero a stringa e ritorno

di C. Graziano

Questa ricetta ha lo scopo di prendere un doppio piccione con una fava, quello di compattare dati su disco e di cavarsi lo sfizio di imparare qualcosa forse utilizzabile altrove. Ma eccoci al trucco, che consta di due routine, la prima per trasformare un dato numerico in una stringa lunga la metà, mentre la seconda compie l'operazione inversa. Eccole.

```
10 REM TRASF. DELLA VAR. A NELLA STRINGA F$
20 F$ = " " : B$ = STR$ (A)
30 C = LEN (B$) : B$ = MID$(B$,2,C)
40 D = INT(LEN(B$)/2)
50 IF D < LEN(B$)/2 THEN B$ = "0" + B$
60 FOR I = 1 TO LEN(B$) STEP 2
70 X = VAL(MID$(B$,I,2))
80 S$ = CHR$ (X) : F$ = F$ + S$
90 RETURN
```

Questa prima routine opera sulla variabile *A* supposta preesistente e relativa al dato da compattare. Alla riga *20* viene trasformata nella stringa *B$*. Alla riga *30* se ne determina la lunghezza (che è poi quella del dato numerico), mentre la parte residua della riga stessa è facoltativa, e serve su alcuni microcomputer che in tale conversione aggiungono uno spazio extra sulla sinistra che va perciò eliminato.
Le due linee seguenti aggiustano la lunghezza di *B$*, in modo che sia formata da un numero pari di caratteri, aggiungendo, se caso, uno zero non-significativo.
Con il ciclo successivo si prelevano a due a due i caratteri di *B$* e si convertono nel valore *X*, che a sua volta, in riga *80*, viene mutato nella stringa ad un sol carattere ASCII *S$*! Trattandosi di coppie di cifre decimali, ossia variabili da 00 a 99 non vi sono problemi in questo compattamento (vedi a proposito una tabella degli ASCII code). Così la stringa *F$* che si va creando nel loop è lunga esattamente la metà di quella di partenza *B$*.
Ancora più corta la routine inversa:

```
10 REM TRASF. STRINGA F$ IN UNA VARIABILE A
20 B$ = " "
30 FOR I = 1 TO LEN(F$)
40 A = ASC (MID$(F$,I,1)) : A$ = STR$(A)
50 B$ = B$ + A$: NEXT: A = VAL (B$)
60 RETURN
```

che si commenta da sé (basta ricordare che *ASC* converte in codice ASCII). le due routine servono a risparmiare spazio su disco, specie se si debbono memorizzare molti dati numerici. Un disco da 100 K contiene circa 15.000 numeri da 6 cifre: con tali routine si arriva ad almeno 25.000 (*vi sono però controindicazioni?* NdR).

La virgola, questa sconosciuta ...

di F. Kalin

Se per un punto martin perse la "K", per una virgola diversi personal tra cui Apple e PET perdono invece frasi intere. Provare per credere. Il fatto (sconcertante e a volte disarmante) si deduce da questi esempi d'uso dei file (dischi o nastri):

```
100 A$ = "AAA,BBB"
110 PRINT # 1,A$
...
...
200 INPUT # 1,A$
210 PRINT A$
```

Il risultato dell'esecuzione della linea *210* è *AAA* (la restante parte della stringa, dalla virgola in poi, è scomparsa: forse che le memorie di massa debbano essere alimentate, oltre che dall'ENEL, anche dalle ben più innocue virgole? ...).
Il baco sembra vivere nell'ombra, ma uno spiraglio di luce si intravede quando l'*INPUT* viene fatto da tastiera: in questo caso appare la scritta *EXTRA IGNORED*, che se da una parte spiega l'arcano (la virgola è usata dal BASIC, anche quello dell'Apple, per consentire di introdurre più dati con una sola istruzione del tipo *INPUT A,B,C,*), dall'altra non estingue il problema.
A questo proposito, i manuali sembrano dire: "Non usate le virgole, ed il problema non sussiste"; ed infatti pare addirittura che gli americani abbiano perso l'abitudine di separare le migliaia con le virgole: come però è possibile che questo consiglio venga accettato da chi, dattilografaro computerizzato, vorrebbe usare il personal per le sue esigenze epistolari?
La soluzione che ho escogitato per il PET, senz'altro una tra le molte possibili, consiste nel sostituire tutte le virgole con un altro simbolo, meglio accetto a dischi e nastri. La routine prescelta è la seguente:

```
10000 DO$=RIGHT$(PR$,1): PR$ = LEFT$(PR$,1)
10010 W = 0:Y = LEN(S$):FOR Z = 1 TO Y
10020 IF MID$(S$,Z,1) = PR$ THEN W = Z:Z = Y
10030 NEXT Z:IF W = 0 THEN RETURN
10040 IF W = 1 THEN B$ = RIGHT$(S$,Y-1):S$ = " ":
```

```
         GOTO 10070
10050    IF W = Y THEN B$ = " ":S$ = LEFT$(S$,Y-1):
         GOTO 10070
10060    B$ = RIGHT$(S$,Y-W):S$ = LEFT$(S$,W-1)
10070    S$ = S$ + DO$ + B$:GOTO 10010
```

nella quale la prima linea ha il compito di preparare i due caratteri, detti *PRIMA* e *DOPO*, in modo che *DOPO* sostituisca *PRIMA*, la terza quello di cercare il carattere da sostituire, le linee *5-6-7* quello di preparare le due parti della linea che precedono e seguono tale carattere, ed infine l'ottava il compito di effettuare la ricostruzione della linea.
Il ritorno della routine avviene solo quando non esistono più caratteri da sostituire (linea *4*).
L'uso della routine richiede che la linea da elaborare sia contenuta nella variabile *S$* e che l'uso della variabile *B$* non provochi malfunzionamenti di altre parti del programma nel quale è inserita.
La chiamata avviene in questo modo:

```
PR$ = ",@" :GOSUB 10000
```

e il risultato è la sostituzione della virgola con il simbolo @; ovviamente in fase di caricamento si dovrà eseguire l'operazione inversa e cioè:

```
PR$ = "@," :GOSUB 10000
```

che restituirà alla linea di testo la sua forma originaria.
E concludo qui, anche se molte altre parole (e non tutte ripetibili) potrebbero essere dette sull'uso dei file ma soprattutto su quello dei manuali.

Blocco Autostart

di G. Minicucci

Vi sono programmi che entrano in esecuzione subito dopo il *LOAD* (a causa dell'istruzione *RUN* inserita all'inizio del listing) e vanno ad eseguire una routine in linguaggio macchina nella quale non è previsto il rientro al BASIC. In questo caso, dopo il *LOAD*, non si riesce a fermare l'esecuzione né andare a vedere il listato, per modificarlo. Con la semplice routine in linguaggio macchina che propongo si pone rimedio alla cosa:

```
1 REM 1234
2 POKE 16514,55
3 POKE 16515,195
4 POKE 16516,67
5 POKE 16517,3
6 FAST
7 RAND USR 16514
```

Dopo che si sono scritte le sette righe, dare il *RUN*, attivando così la routine di *LOAD*; quindi accendere il registratore, al che il calcolatore si blocca segnalando errore c/7. Con questo trucco l'autostart è inibito e sarà possibile leggere/variare il listato. L'Assembly e l'esadecimale sono:

```
37          SCF          ; accendi il flag del carry
C3 43 03    JP 0343
```

Save Repeat

di M. Manna

Per ottenere la ripetizione del salvataggio di un programma aggiungere le seguenti linee in fondo al relativo listato:

```
9990  STOP
9992  CLEAR (se non sono necessarie le variabili)
9993  LET N$ = <nome programma >
9994  FOR F = 1 TO 3 (numero di volte che si vuole fare il
      SAVE)
9995  PRINT AT 10,0; F "SAVE"; N$; TAB 0; "R PER RUN
      PROGR."
9996  INPUT A$
9997  IF A$ = "R" THEN RUN
9998  SAVE N$
9999  NEXT F
```

Quando il programma verrà ricaricato e si stamperà il nome di esso, battere *R* in *INPUT* per far partire l'esecuzione.

Rilocare l'Apple BASIC e altre cosucce

di E. M. Staderini

È abbastanza noto che i programmi BASIC Applesoft vengono allocati in memoria RAM a partire dalla locazione $801 (decimale 2049), anche se è estremamente importante che la locazione $800 sia azzerata. Di qui discende che la seconda pagina di testo, pur disponibile sul video, non possa essere completamente utilizzata proprio perchè si sovrappone in parte o in tutto con il programma BASIC.
Bene, ecco arrivato il momento di far partire i programmi BASIC da una locazione più alta, così da liberare completamente la seconda pagina di testo. A tale scopo basterà che i programmi vengano memorizzati a partire dalla locazione $C00. La routine *PROH* riportata in figura 1 equivale ad un comando *NEW* del BASIC, tranne che, dopo essere stata eseguita, i programmi verranno memorizzati a partire proprio dalla locazione $C00; il bello di tutta la faccenda è che pure i programmi caricati da disco (e costruiti in precedenza a partire dalla locazione $800) verranno ora rilocati e avranno inizio da lì.
Tutto andrà come prima però resterà disponibile la RAM della seconda pagina di testo.
Non si ha modo di rimettere le cose a posto (neanche con *RESET*) se non mettendo in esecuzione la routine *PROL* di figura 2 la quale pure funge da comando *NEW*, però d'ora in poi i programmi saranno fatti partire dalla locazione $800.
Queste routine a circonfuse di magia, in realtà funzionano molto semplicemente andando ad alterare opportuni puntatori in pagina zero, e precisamente:

- $0067-$0068: puntatore all'inizio del programma ($801 oppure, nel nostro caso, $C01);
- $0069-$006A: puntatore all'inizio dello spazio per variabili semplici (*LOMEM*);
- $006B-$006C: puntatore all'inizio dello spazio per gli array;
- $006D-$006E: puntatore all'inizio dello spazio libero;
- $00AF-$00B0: puntatore alla fine del programma.

Resta ora da vedere come utilizzare la seconda pagina di testo resa disponibile.
Intanto si notino le due routine *TEXT2* (figura 3) e *TEXT1* (figura 4) che servono rispettivamente per passare al display pagina 2 ed al display pagina 1.
Purtroppo non è possibile scrivere, né con istruzioni BASIC, né con istruzioni del monitor, sulla seconda pagina di testo e questo perchè la routine del monitor che calcola l'indirizzo base per la scrittura è prevista solo per la pagina 1.
È un vero peccato se si pensa che i ritocchi da apportare al monitor per ovviare a tale manchevolezza sarebbero semplicissimi solo che esso risiede su ROM ed è cioè l'intoccabile.
A meno di progettare all'uopo del software opportuno (copiando dal monitor stesso), il modo più semplice di utilizzare la seconda

pagina di testo consiste nel ricopiarci sopra la prima con una routine molto veloce quale è la *TX 1→2* riportata in figura 5; l'inverso può essere fatto con la routine *TX 2→ 1* riportata in figura 6.
A questo punto la palla del gioco passa all'inventiva di chi legge, comunque una possibile utilizzazione potrebbe essere la seguente: supponendo di dover sottoporre all'utente una serie di videate di testo senza voler far vedere il processo di scrittura, c'è da agire così: mettere su video la pagina 2 e intanto scrivere sulla 1; quindi trasferire la 1 sulla 2, cancellare tutto sulla 1 e, mentre l'utente sta leggendo sulla 2, preparate di nuovo la 1 che poi sarà ritrasferita sulla 2 e via di seguito.
Questo "page flipping" darà l'effetto di una velocissima dissolvenza incrociata che conferirà un che di professionale alle nostre videate.
Tutti i programmini descritti e riportati nelle figure sono completamente rilocabili senza problemi ed anche evocabili da BASIC con delle *CALL* i cui argomenti, che vanno in decimale, si possono ricavare convertendo gli indirizzi di inizio delle varie routine.

```
0300-   A9 00       LDA   #$00
0302-   8D 00 0C    STA   $0C00
0305-   8D 01 0C    STA   $0C01
0308-   8D 02 0C    STA   $0C02
030B-   A9 01       LDA   #$01
030D-   85 67       STA   $67
030F-   A9 04       LDA   #$04
0311-   85 69       STA   $69
0313-   85 6B       STA   $6B
0315-   85 6D       STA   $6D
0317-   85 AF       STA   $AF
0319-   A9 0C       LDA   #$0C
031B-   85 68       STA   $68
031D-   85 6A       STA   $6A
031F-   85 6C       STA   $6C
0321-   85 6E       STA   $6E
0323-   85 B0       STA   $B0
0325-   60          RTS
```

Figura 1 - Routine PROH.

```
0326-   A9 00       LDA   #$00
0328-   8D 00 08    STA   $0800
032B-   8D 01 08    STA   $0801
032E-   8D 02 08    STA   $0802
0331-   A9 01       LDA   #$01
0333-   85 67       STA   $67
0335-   A9 04       LDA   #$04
0337-   85 69       STA   $69
0339-   85 6B       STA   $6B
033B-   85 6D       STA   $6D
033D-   85 AF       STA   $AF
033F-   A9 08       LDA   #$08
0341-   85 68       STA   $68
0343-   85 6A       STA   $6A
0345-   85 6C       STA   $6C
0347-   85 6E       STA   $6E
0349-   85 B0       STA   $B0
034B-   60          RTS
```

Figura 2 - Routine PLOL.

```
034C-   AD 51 C0    LDA   $C051
034F-   AD 52 C0    LDA   $C052
0352-   AD 55 C0    LDA   $C055
0355-   60          RTS
```

Figura 3 - Routine TEXT2.

```
0356-   AD 51 C0    LDA   $C051
0359-   AD 52 C0    LDA   $C052
035C-   AD 54 C0    LDA   $C054
035F-   60          RTS
```

Figura 4 - Routine TEXT1.

```
0360-   A9 00       LDA   #$00
0362-   85 42       STA   $42
0364-   A9 08       LDA   #$08
0366-   85 43       STA   $43
0368-   A9 00       LDA   #$00
036A-   85 3C       STA   $3C
036C-   A9 04       LDA   #$04
036E-   85 3D       STA   $3D
0370-   A9 FF       LDA   #$FF
0372-   85 3E       STA   $3E
0374-   A9 07       LDA   #$07
0376-   85 3F       STA   $3F
0378-   A2 00       LDX   #$00
037A-   A1 3C       LDA   ($3C,X)
037C-   81 42       STA   ($42,X)
037E-   20 B4 FC    JSR   $FCB4
0381-   90 F7       BCC   $037A
0383-   60          RTS
```

Figura 5 - Routine TX1→2.

```
0384-   A9 00       LDA   #$00
0386-   85 42       STA   $42
0388-   A9 04       LDA   #$04
038A-   85 43       STA   $43
038C-   A9 00       LDA   #$00
038E-   85 3C       STA   $3C
0390-   A9 08       LDA   #$08
0392-   85 3D       STA   $3D
0394-   A9 FF       LDA   #$FF
0396-   85 3E       STA   $3E
0398-   A9 0B       LDA   #$0B
039A-   85 3F       STA   $3F
039C-   A2 00       LDX   #$00
039E-   A1 3C       LDA   ($3C,X)
03A0-   81 42       STA   ($42,X)
03A2-   20 B4 FC    JSR   $FCB4
03A5-   90 F7       BCC   $039E
03A7-   60          RTS
```

Figura 6 - Routine TX2→1.

Si, ma il Pascal come si avvia?

di **G. Ugolini**

Pronto, parlo con ... (il mio rivenditore di fiducia) sono Ugolini; senta è vero che in Pascal si possono fare delle animazioni molto particolari?
Interlocutore: ma certo, è stupendo ... (e via magnificando).
Va bene, allora ordini per favore la scheda Pascal.
Circa dieci giorni dopo la scheda sospirata è arrivata ed eccomi trepidante che mi accingo ad entrare nel mondo del Pascal UCSD.
Erano tre i manuali forniti in dotazione. Non ricordo con quale iniziai, ma certamente con quello che spiegava come fare le copie dei dischi con un unico drive. Ahimé, malgrado tutti i miei sforzi, non cavai un ragno dal buco: non sapevo come infilare un programma ed eseguirlo.
In BASIC si scrive *NEW* si preme *RETURN*, quindi si scrive *10 PRINT "SALVE COME VA"* si ripreme *RETURN* poi si fa *RUN* seguito da *RETURN* e il programma è a posto, ma in Pascal?
Guardai gli esempi, provai a seguirli, succedeva di tutto meno quello che volevo.
Telefonai persino a Milano per cercare di carpire qualche notizia precisa su come si inizia un programma, ma non ci fu verso.
Alla fine andai a Genova e copiai, tasto per tasto, tutto quanto un mio amico faceva.
Il primo programma che gli vidi inserire riguardava la grafica.
Volevo solo che il computer disegnasse due righe e poi avrei potuto morire felice.
Mi ricopia l'intera sequenza dei tasti da battere, come ho già detto e in tal modo, bene o male riuscii con mia somma soddisfazione a far girare il programma. Mi sentivo con il vento in poppa e mi pareva di avere il mondo in tasca.
Oggi leggo Apple Pascal, A. Hands-On Approach (Arthur Luhermann della Mc Graw Hill $ 14.95).
A quanti hanno la tentazione di buttare la scheda-linguaggio e scambiare i manuali con un astuccio di matite e temperino, consiglio di rinfrancarsi e di soprassedere da sì tragiche determinazioni.

Come si inserisce un programma in Pascal

La sequela che sopra accennavo si riassume così.
Spegnere l'Apple e inserire il disco Apple 3.
Accendere l'Apple: il disco ruota e compaiono dei segni; quando il drive si ferma e compare un quadratino in alto a sinistra togliere il disco Apple 3 e inserire il disco Apple 0, quindi si prema *RESET* (se non succedesse niente *CTRL-RESET*).
Dopo ulteriori giri del disco e altri segni strani, da ultimo compare:

```
COMMAND: E(DIT, R(UN, F(ILE, C(OMP, L(IN
   WELCOME APPLE 0 TO ... bla bla bla
```

Qui si premerà *E* che sta per Editor, quindi *RETURN*: comparirà > *EDIT: A(DJST C(PY D(LETE F(IND I(NSRT* premere *I* che sta per Insert comparirà > *INSERT: TEXT [<BS> A CHAR, <DEL> A LINE]*.
Scrivere (immaginando un titolo scherzoso per il primo, sospirato, programma):

```
PROGRAM FINALMENTE; seguito da RETURN
BEGIN (ret)
WRITE ('FINALMENTE') (ret)
END. quindi battere CTRL-C
```

Adesso sono questi i tasti che in successione vanno pigiati: *Q U R* cioè *Quit Update Run*.
Il disco si metterà in moto, compariranno scritte varie poi, all'improvviso, il buio dello schermo e quello della tua vita saranno squarciati dalla scritta

```
COMMAND: E(DIT, R(RUN, F(ILE, C(OMP, L(IN
RUNNING .... (ce l'hai fatta)
FINALMENTE
```

Premendo *R* il programma rivà!
Se si vuol scrivere un altro programma quando compare la scritta *COMMAND*: bla bla bla si prema il tasto *F* poi *N* poi *Y Q* e ci si ritroverà daccapo (premere *E, RETURN, I,* scrivere *PROGRAM* e via, di nuovo, come sopra).
Il Pascal non è tutto qui, c'è dell'altro naturalmente. Ma tutto sta ad incominciare, se no che gusto c'è?

La sfida del mese

Inauguriamo qui, in quest'ambito culinario, l'abitudine del concorsino e dello challange. Stavolta lo poniamo noi direttamente, però in seguito potrebbe farlo chicchessia e, in caso di pubblicazione dell'arduo-ma-no non-troppo (possibilmente) interessante quesito gli verrebbe attribuito il compenso standard della rubrica "Il Ricettario".
Agli eventuali solutori particolarmente brillanti potranno essere assegnati premi sotto forma non di salsicce o caciocavalli, ma di abbonamenti a **Bit** (l'equivalente in libri Jackson).
Prima che mi dimentichi: chi invia una sfida mandi pure la soluzione, scanso nostro rischio di ricovero al Neurodeliri (alias ammattimento).
Il problemino con cui iniziamo, tanto per dare il la, è semplice, ma forse gli spunti che suggerisce non sono del tutto banali, almeno per i meno scafati.
Supponiamo di avere una subroutine, già nota e collaudata, che opera su un generico vettore o array (per fissare le idee: una classica routine di ordinamento): chiamiamolo *XA*. Si desidera poterla utilizzare chiamandola da più punti di un certo main program.
Supponiamo inoltre che vorremo passare alla nostra brava procedura, di volta in volta, un diverso array, a guisa di parametro. Prima di discutere la questione, ricordiamo come ci si comporta quando si tratta di una comune variabile: sia *X* il parametro e la routine parta da riga *1000*; per operare con le variabili *A*, *B*, ecc. faremo precedere rispettivamente l'istruzione *X = A*, *X = B* ecc. prima della chiamata *GOSUB 1000*.
In modo analogo al rientro della subroutine, però per semplicità *non* ci occupiamo di questo secondo aspetto, sostanzialmente speculare. Se adesso volessimo seguire lo stesso procedimento, pedissequamente, nel caso di array occorrerebbe far precedere una riga del tipo:

```
FOR K = 1 TO N: X(K) = A(K): NEXT(N = dimensione di A)
```

Chiaramente si tratta di soluzione abbastanza onerosa in termini di velocità e spazio (il che per routine corte vanifica il vantaggio di non doverle ripetere più volte, anche se questo spesso si fa ugualmente, per strutturale chiarezza).
Entrambi gli svantaggi sono più cospicui con vettori a più dimensioni (e si tenga conto che l'array *X* deve esser dimensionato con potenza uguale a quella del maggiore tra quelli in gioco nel "principale").
Una soluzione alternativa potrebbe consistere, sempre in BASIC, nell'uso di opportuni *POKE*, sempre a monte della *GOSUB 1000*: si tratta di "ficcare" brutalmente in tutti i punti in cui figura un elemento generico *XA(I)* e *XA(K)*, con *I* e *K* ovviamente qualsiasi, l'elemento specifico che in quella chiamata ci serve. In tal modo l'occupazione di memoria varia ma può spesso essere onerosa, così pure

risultare eccessiva la perdita di tempo.
Ma il guaio peggiore è che tale tecnica, oltre a richiedere una certosina pazienza ed una precisa conoscenza del proprio sistema è scarsamente portabile: sia da macchina a macchina, sia in occasione di modifiche al nostro stesso programma, tali da comportare variazioni dalla collocazione "fisica" delle istruzioni, diciamo così, logiche.
Per pura curiosità, poniamo una prima minisfida a chi riuscisse in qualche modo a rendere più portatile questo metodo.
Dimenticavamo di dire che l'unico suo vantaggio risiede nel fatto che, se si riflette, *XA* è una sorta di "array-fantasma" e quindi *non* occupa memoria extra.
Buffo, vero?
Orbene diamo qualche minuto per riflettere e diciamo che qualcosa di equivalente può ottenersi anche senza ricorrere agli allegri *PEEKE* & *POKE*. La soluzione è riportata qui sotto capovolta.
Siccome, come si vedrà, è ancora imperfetta ecco la:
SFIDA DEL MESE: suggerire un miglioramento atto a contenere al minimo lo spreco di memoria per gli array. ■

Soluzione

È un ovino di Colombo (anche facilino, peraltro). Noi stiamo parlando di array ad *una* dimensione, ma chi ci vieta di considerare il loro assieme come un array-of-array (come si dice in Pascal) ossia bidimensionale?
Anziché chiamarli, poniamo, *A1*, *A2*, ... *Ak* ... *An*, essi costituiranno un unico *A(i,k)* i cui esemplari monodimensionali saranno: *A(i,1)*, *A(i,2)* ... *A(i,k)* ... *A(i,n)*. La "sub" agirà sul generico *A(i,k)* e, prima di invocarla, le verrà passato l'opportuno parametro K (variabile da 1 a *n*) che individua il *particolare* insieme, unitamente alla dimensione *M* del medesimo.
Qual'è però l'inconveniente? Specie con array di lunghezza molto eterogenea il dimensionamento di *A(i,k)* dovrà prevedere una (comune) dimensione, diciamo così, "orizzontale" pari a quella massima, cioè dell'array maggiore. Conseguenza: spreco di spazio.
Come rimediare a quest'ultima imperfezione?

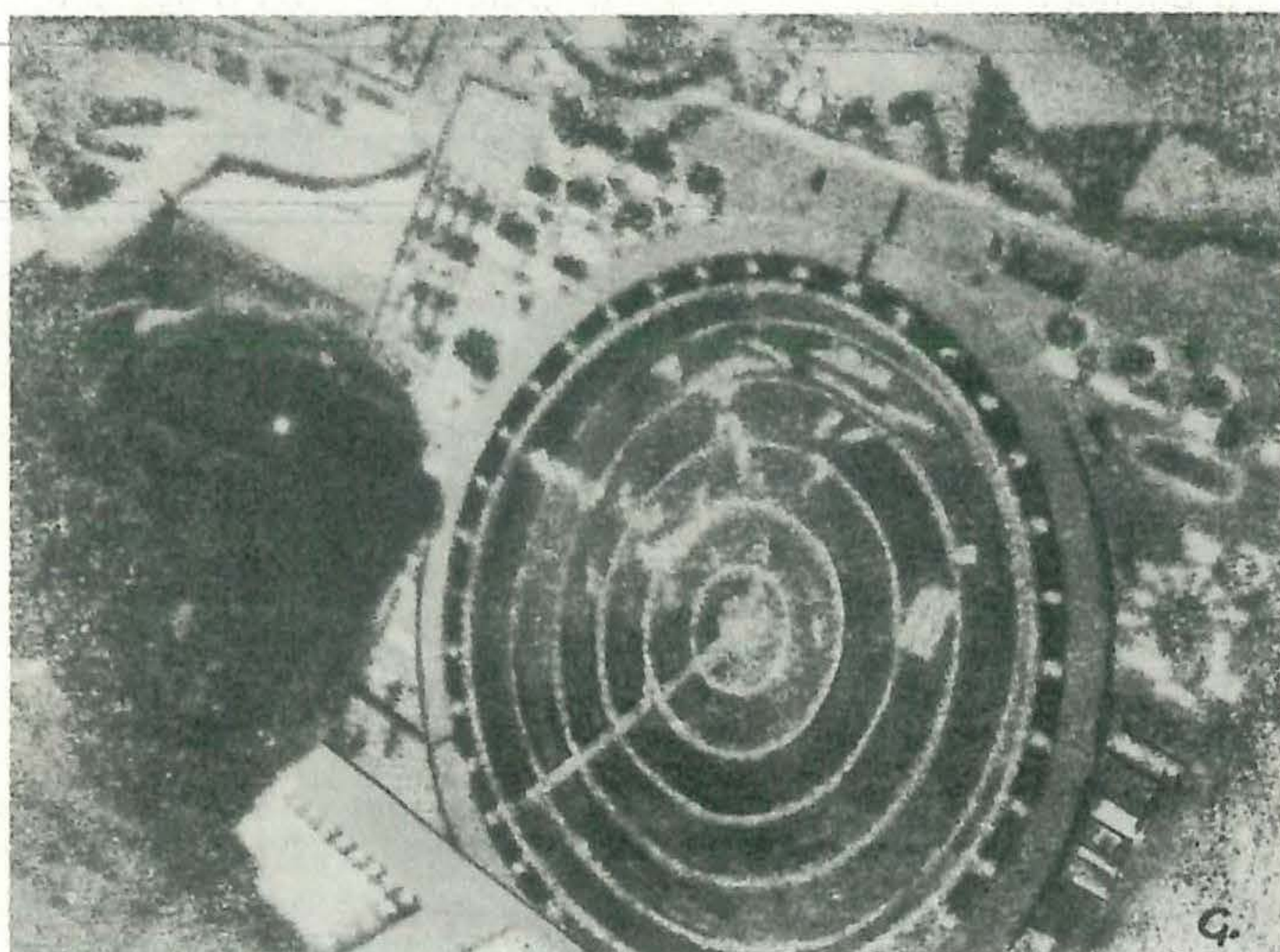
Software in vetrina

GESTIONE FINANZIARIA PERSONALE

Un sofisticato programma di gestione finanziaria (fino a lire 99.999.999) con funzioni di bilancio differenziato per singole voci (sia per le entrate che per le uscite), singoli bilanci (fino a 26 argomenti), resoconti parziali e totali. Le voci in ingresso ed uscita possono essere periodiche (mensili) o uniche (annuali). Il programma gestisce la funzione di memorizzazione su nastro per le funzioni di archivio sia funzionale che storico.

Cassetta TF/0102-12
L. 22.000 IVA esclusa
ZX81 - 16 Kbyte RAM

DOWN UNDER

Il piccolo canguro Joey è in difficoltà, non riesce a saltare nella borsa della madre. Aiutatelo voi. Battete ogni volta un numero compreso fra 1 e 10 che corrisponde alla distanza del piccolo dalla madre.

SUBMARINES

Siete il capitano di una nave e dovete distruggere tutti i sottomarini presenti in uno specchio di mare. Quando venite invitato a sparare inserite il numero delle coordinate dove pensate si trovi il sottomarino.

DOODLING WITH GRAPHICS

Disegnate con i caratteri grafici sullo schermo. Il programma sceglie da solo il primo simbolo, ma potete cambiarlo scegliendo voi il simbolo che volete.

THE INVISIBLE INVADERS

In quante mosse riuscirete a catturare e distruggere l'invasore proveniente dalla Via Lattea, con la vostra astronave guidata solo dal radar che vi dà la direzione da prendere?

REACTION

Misurate la velocità dei vostri riflessi contro i vostri amici. Sul video viene visualizzata una lettera, dovete premere questa lettera più velocemente possibile. Il computer valuterà la vostra velocità di risposta. Alla fine vedrete il vostro punteggio.

PETROL

Effettua facili conversioni sul consumo di carburante di un'automobile passando da numero di miglia per gallone a numero di litri per 100 chilometri o viceversa.

Cassetta TF/0100-56
L. 35.000 IVA esclusa
ZX81 - 1 Kbyte RAM

VENTUNO

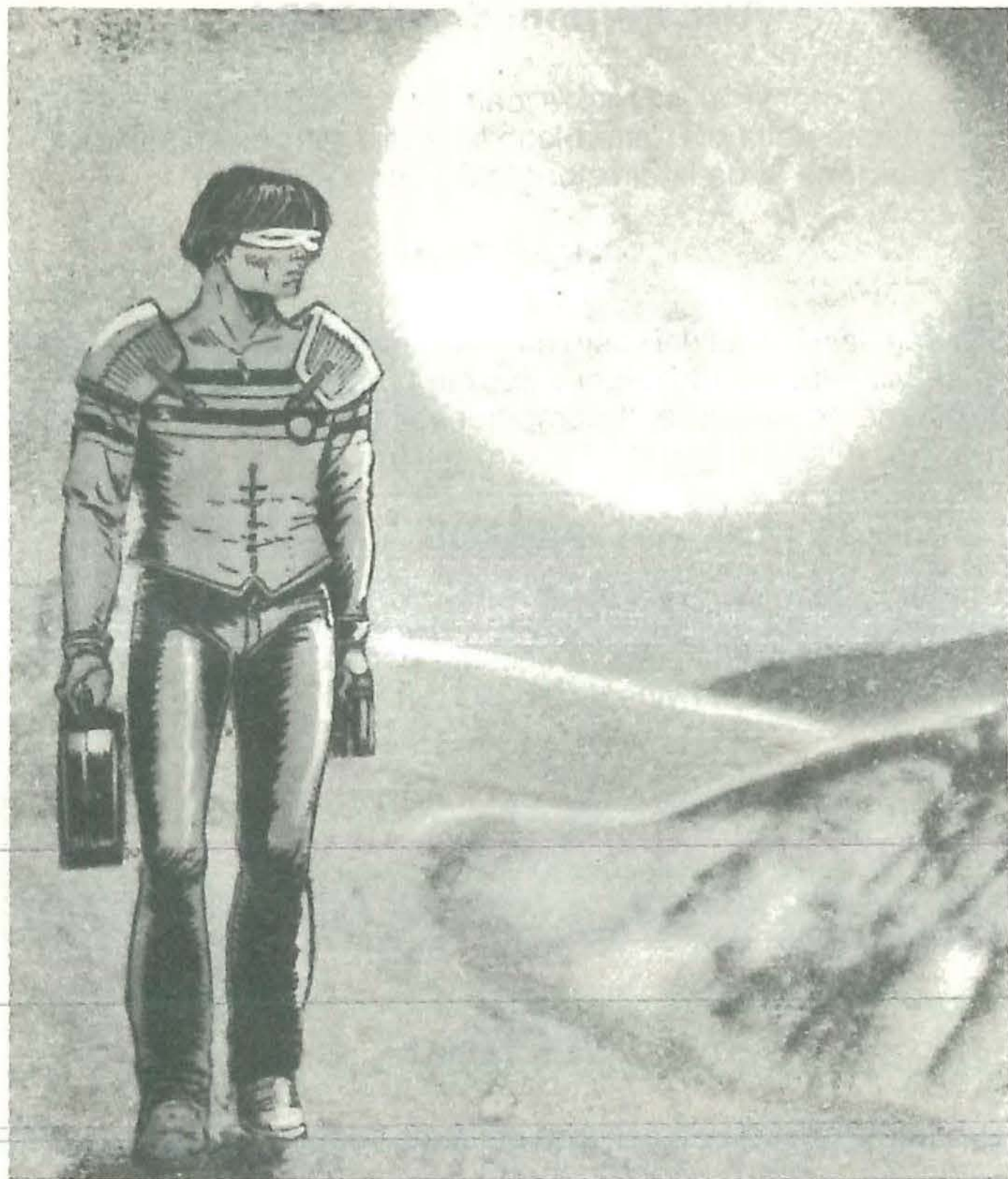
Un interessante gioco con il calcolatore. *Ventuno* è un gioco con le carte; per vincere bisogna ottenere un punteggio fra 16 e 21. Il Sinclair fa da banco e distribuisce le carte per sé e per il giocatore. A parità di punteggio vince il banco ma il giocatore ha diritto a scegliere la puntata. Dopo ogni carta il giocatore può scegliere se fermarsi, se voltare un'altra carta o se scommettere ancora. Si possono voltare al massimo 5 carte, ma se un giocatore riesce a voltarne 5 senza "*SALTARE*" (cioè senza superare il punteggio di 21) quale che sia il punteggio quella mano è considerata vincente. L'unica mano superiore alle 5 carte è il 21 in 2 sole carte (realizzabile con una figura e un asso). Se un giocatore riesce a realizzarla il banco paga doppio.

MICROCALC

Microcalc è un programma di applicazione scientifica, consente infatti di memorizzare fino a 26 espressioni di 15 caratteri. Si noti che le funzioni matematiche del Sinclair, come *SIN, COS, EXP* ecc., contano come un carattere. Ogni espressione è indicata con una lettera e può essere costituita da una costante o da operazioni anche su altre variabili. È possibile salvare su un nastro una certa situazione. Una volta che il programma è stato salvato esso conserva anche tutti i dati che conteneva il Microcalc.

Cassetta TF/0103-32
L. 17.000 IVA esclusa
ZX81 - 16 Kbyte RAM

GEOGRAPHY 1

TOWNS IN ENGLAND AND WALES
Il calcolatore vi fa vedere una mappa dell'Inghilterra e del Galles. In seguito vi viene chiesto di indicare su una carta geografica, in cui mancano delle città inglesi o gallesi, l'esatta collocazione di quelle appartenenti all'elenco che vi viene fatto vedere. Un nuovo modo per avere maggiori notizie su un argomento che vi interessa.

COUNTRIES AND CAPITALS OF EUROPE
Le identiche modalità del gioco sulle città con la differenza che in questo caso si tratta di capitali e stati d'Europa.

Cassetta TF/0100-22
L. 35.000 IVA esclusa
ZX81 - 16 Kbyte RAM

CREA - LISTA

Finalmente potete sbrigare la corrispondenza molto rapidamente con il calcolatore. Due utili programmi, *CREA* e *LISTA* che consentono di creare e gestire testi e archivi di indirizzi in ordine alfabetico per cognome-nome, che vi aiuteranno nella gestione della corrispondenza di affari o privata.
Il MENU del programma *CREA* è il seguente:

1. CREO UN TESTO
2. CREO INDIRIZZI
3. AGGIORNO INDIRIZZI
4. AGGIORNO TESTO
5. FINE

È possibile creare un testo di 70 caratteri per riga e tale formato viene mantenuto in fase di stampa. Questo viene memorizzato su cassetta e può essere aggiornato, cioè si possono cancellare righe, aggiungere o modificare quelle già esistenti. Stessa cosa vale per l'archivio indirizzi che vengono caricati in ordine alfabetico per cognome-nome.
Il MENU del programma *LISTA* è il seguente:

1. LISTO UN TESTO
2. LISTO INDIRIZZI
3. STAMPA CIRCOLARI
4. STAMPA ETICHETTE
5. FINE

È possibile listare testi o indirizzi sia sul video che sulla stampante. Le lettere (circolari) possono essere inviate a tutti i nominativi di un archivio indirizzi o a partire da un determinato nominativo. Il testo di una lettera non deve superare le 60 righe (di 70 caratteri ciascuna). Le modalità di stampa delle lettere sono le seguenti:

A) Testo della lettera non personalizzato.
B) Lettera iniziante con una frase richiesta a programma, tipo Egr. Sig. Nome Cognome, seguita dal testo.
C) Lettera iniziante con una frase richiesta a programma, tipo Mio caro Nome, seguita dal testo.
D) Indirizzo completo in alto a destra:

NOME COGNOME
INDIRIZZO
CAP CITTA'

e lettera iniziante, in uno dei tre modi visti prima, ma in posizione tale che il foglio possa essere piegato in 3 parti, graffettato e spedito senza busta.
Le etichette indirizzo vengono stampate affiancate a due a due. Con questi programmi sono trattabili archivi contenenti un massimo di 30 indirizzi per ciascuno. Se il volume di indirizzi che si vuole trattare è molto ampio, si possono creare più archivi.

Cassetta TF/9350-02
L. 60.000 IVA esclusa
VIC 20 con espansione da 16 Kbyte + Tape + Printer

INVENTIONS 1

INVENTIONS BEFORE 1850
Il calcolatore organizza una competizione per un massimo di 4 giocatori. Si tratta di rispondere alla domanda che viene posta dal calcolatore riguardante una delle 7 voci a scelta su argomenti che riguardano invenzioni ed inventori prima del 1850. Per ogni risposta esatta aumenta il vostro punteggio in caso contrario torna indietro. Un nuovo modo per avere maggiori notizie su un argomento che vi interessa.

INVENTIONS SINCE 1850
Le identiche modalità del gioco precedente con la differenza che in questo caso si tratta di invenzioni ed inventori dopo il 1850.

Cassetta TF/0100-36
L. 35.000 IVA esclusa
ZX81 - 16 Kbyte RAM

HISTORY

EVENTS IN BRITISH HISTORY
Si tratta di rispondere alla domanda che viene posta dal calcolatore riguardante uno dei 4 periodi a vostra scelta su argomenti che riguardano eventi e fatti della storia britannica. Una volta selezionato il periodo dovete scegliere il modo d'uso: insegnamento casuale o cronologico oppure gara fra uno o più giocatori. Per ogni risposta esatta aumenta il vostro punteggio storico in caso contrario diminuisce. Un nuovo modo per avere maggiori notizie su un argomento che vi interessa.

KINGS
Le identiche modalità del gioco precedente con la differenza che in questo caso si tratta di domande riguardanti fatti e nomi di monarchi britannici e non di eventi storici.

Cassetta TF/0100-26
L. 35.000 IVA esclusa
ZX81 - 16 Kbyte RAM

SPECTACULATOR

Spectaculator è un sofisticato foglio elettronico di al massimo 99 righe e 99 colonne con 27 caratteri utilizzabili in ogni casella. Su questo si possono eseguire diversi tipi di calcoli che vanno dalla gestione del bilancio familiare al calcolo di problemi matematici e geometrici o altro. Lo schermo presenta una matrice vuota e il calcolatore è in attesa di un comando. Premete il tasto ? per avere la lista dei comandi utilizzabili che sono:

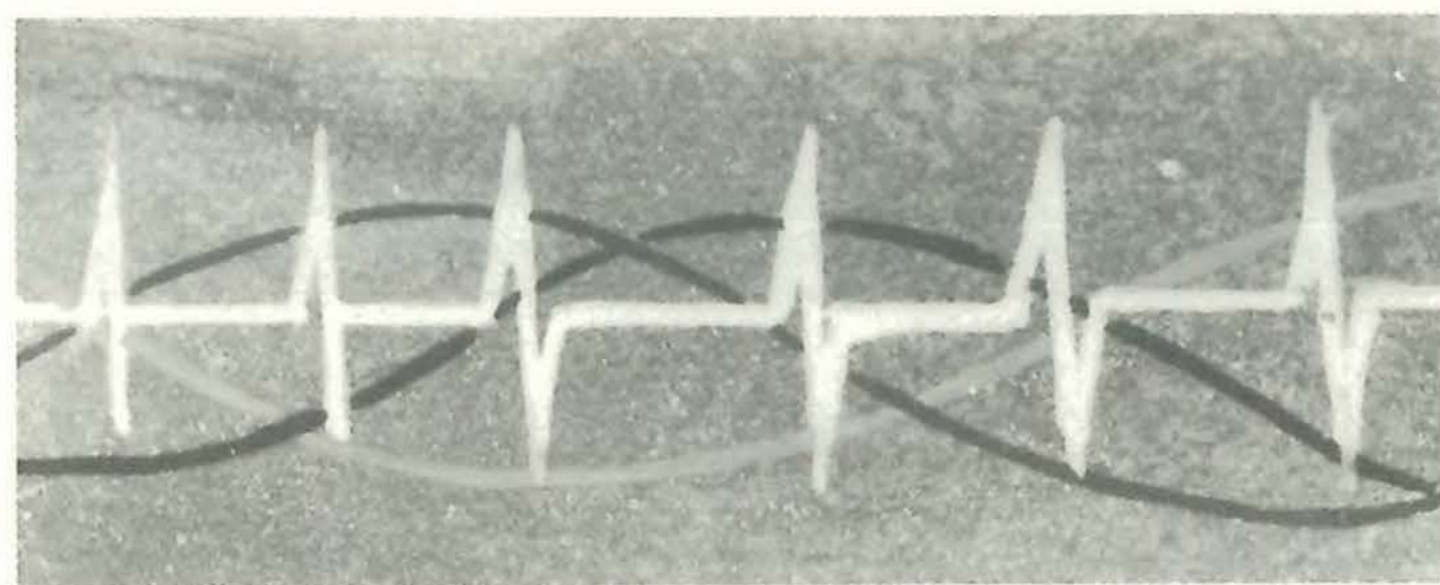
MM Muove il cursore
EN Inserisce numeri
CF Inserisce una formula che opera sulle colonne
RF Inserisce una formula che opera sulle righe
CA Svolge i calcoli
ET Inserisce un testo
CR Pulisce la riga
CC Pulisce la colonna
CW Cambia la lunghezza dei dati inseribili nelle colonne
FR Visualizza la memoria disponibile
DR Cancella una riga
DC Cancella una colonna
IR Inserisce una riga
IC Inserisce una colonna
SA Memorizza la matrice sul registratore a nastro
LO Carica una matrice dal registratore
LI Lista sulla stampante

CHIAVI SPECIALI:

BREAK Torna in attesa di un comando
CLEAR Torna indietro di una posizione
? Propone la lista di quello che potete fare col comando che state usando. Sono disponibili 3 funzioni:
SUM Fa sommare al calcolatore i valori dalla colonna specificata a quella dove si trova il cursore.
SMT Dà i totali cumulativi ed il totale finale di una specificata colonna.
SQR Per la radice quadrata.
Per qualunque calcolo è possibile specificare prima se il valore deve essere decimale o intero. I calcoli programmati sulle righe oppure sulle colonne, non agiscono quando trovano una casella già occupata da una formula.

Cartuccia TF/3104-00
L. 90.000 IVA esclusa
TRS80 - Color

BIORHYTHMS 1

WHAT IS BIORHYTHMS?
Questo programma dà la spiegazione di cosa sono i bioritmi indicandoci come leggere i grafici che ogni volta vengono disegnati sullo schermo.
Dando una spiegazione di cosa sono i 3 cicli e la loro durata *P* (23 giorni) *E* (28 giorni) *I* (33 giorni). Ogni ciclo sia fisico, emotivo che intellettuale ha dei giorni critici, massimi e minimi. I giorni visibili ogni volta sono 32 e cominciano con il giorno della data odierna (today's date).

YOUR BIORHYTHMS
Il programma disegna sullo schermo i vostri bioritmi. Segnalandovi i punti critici di massima e di minima per ogni ciclo. Potete inserire date a vostro piacimento, una per volta, per avere più grafici per diversi periodi, o per diverse persone. Con un'apposita funzione potete vedere i vostri giorni critici e le percentuali sui 3 cicli in un dato giorno; variabili fra + 100 (massimo) 0 (giorno critico) e − 100 (minimo). Inoltre potete ricercare in un certo periodo di tempo quando si verificano speciali condizioni, da voi definte.

Cassetta TF/0100-20
L. 35.000
ZX81 - 16 Kbyte RAM

HI RES

Routine in linguaggio macchina per applicazioni grafiche ad alta risoluzione della ZX Printer.
Il programma *HI RES* se caricato prima della stesura di un nuovo programma che richieda output in alta risoluzione, fornisce all'utente i comandi fondamentali per tale tipo di grafica.
Nella "pagina grafica", costituita dalla matrice *A$*, sono indirizzabili 44056 punti, divisi in 256 colonne per 176 righe.
Comando *PLOT*: per stampare un punto di coordinate *X* e *Y*, che viene memorizzato in pagina grafica.
Comando *DRAW*: il comando serve a tracciare rette, con qualsiasi direzione e inclinazione, comprese tra l'ultimo punto precedentemente stampato, e il nuovo punto *X, Y*. Il punto precedentemente tracciato (origine della retta), può essere stato definito sia da un comando *PLOT*, che da un altro comando *DRAW*.
Comando *OUT*: per ottenere la copia su stampante della pagina grafica, che viene stampata senza interruzioni alla massima velocità consentita alla ZX Printer.

ESEMPIO DI PROSPETTIVA

Programma dimostrativo delle possibilità di HI RES.
Il programma fornisce un esempio di rappresentazione su carta di figure tridimensionali, partendo dall'introduzione delle coordinate spaziali (*X, Y, Z*) dell'oggetto, e permettendone la visualizzazione sotto diversi punti di vista. Nel programma sono già contenute le coordinate spaziali di due diversi oggetti, e vengono richiesti all'utilizzatore solo i parametri relativi al punto di osservazione. Il computer consiglia il range dei valori su cui operare.

ESEMPIO DI FUNZIONE

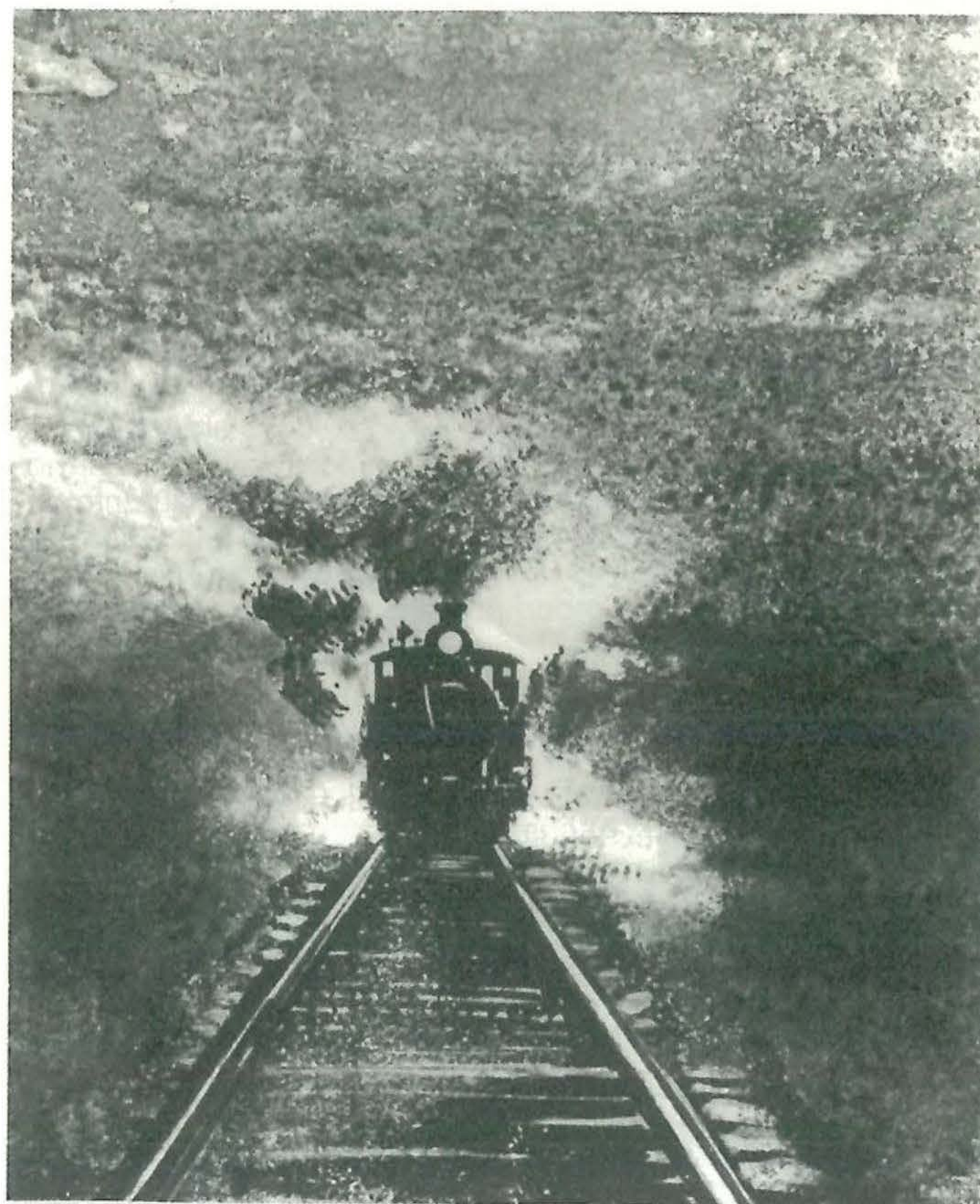
Programma dimostrativo delle possibilità di HI RES
Questo programma realizza un output grafico di una funzione trigonometrica in forma pseudo tridimensionale. Viene costruito un reticolo sul piano *X-Z* (*A* e *B*) con un range per entrambi di + o − *360*. Per ognuno dei 289 punti su *X-Z* viene calcolata la funzione *Y=SIN* R, dove *R=SQR(X 2+Z 2)*.

Cassetta TF/0103-36
L. 22.000 IVA esclusa
ZX81 - 16 Kbyte RAM + Printer

MUSIC 1

COMPOSERS
Un nuovo modo per avere maggiori notizie su un argomento che vi interessa. Il calcolatore organizza una competizione per un massimo di 4 giocatori. Si tratta di rispondere alle domande che vengono poste e che riguardano i compositori. Per ogni risposta esatta aumenta il vostro punteggio musicale, in caso contrario torna indietro.

MUSICIANS
Le identiche modalità del gioco sui compositori con la differenza che in questo caso si tratta di musicisti.

Cassetta TF/0100-28
L. 35.000 IVA esclusa
ZX81 - 16 Kbyte RAM

TRAIN RACE

Siete un giocatore e dovete scommettere su una corsa fra 3 treni. A voi indovinare quale dei 3 vincerà.

CHALLENGE

Dovete indovinare un numero generato dal calcolatore che è compreso in un rango da voi precedentemente definito. Il calcolatore vi aiuta segnalandovi se il numero che gli date è maggiore o minore di quello che lui ha generato. Il gioco finisce quando indovinate il numero, allora il calcolatore vi segnala il numero dei tentativi che avete dovuto fare.

SECRET MESSAGE

Scrivete un messaggio che dovrà essere indovinato dagli altri concorrenti. Il messaggio scorre sullo schermo in maniera confusa per confondere gli avversari.

MIND THAT METEOR

Siete al comando di un'astronave e dovete attraversare una cortina di meteoriti.

CHARACTER DOODLE

Disegnate sullo schermo in tutte le direzioni a vostro piacimento usando caratteri alfabetici.

CURRENCY CONVERSION

Questo programma effettua conversioni di importi da franchi, dollari, lire ecc. a sterline.

Cassetta TF/0100-54
L. 35.000 IVA esclusa
ZX81 - 1 Kbyte RAM

AVENGER

Vi trovate di fronte ad una invasione degli alieni siete da solo e avete con voi tre cannoni laser. Gli alieni scendono verso la base operazionale in gruppi compatti sparando e lanciando bombe. Vengono sbarcati da una astronave che ogni tanto appare in alto sullo schermo. I punteggi per ogni alieno abbattuto sono:

Quinta fila	30 punti
Quarta fila	20 punti
Terza fila	20 punti
Seconda fila	10 punti
Prima fila	10 punti

Il punteggio per l'abbattimento dell'astronave viene assegnato in modo casuale. Se raggiungete 1500 punti vi viene fornito un altro cannone laser. Il punteggio che avete raggiunto viene segnalato in alto sullo schermo con a fianco il punteggio massimo raggiunto nelle partite precedenti. Le istruzioni sono fornite da programma.

Cartuccia TF/9300-20
L. 37.000 IVA esclusa
VIC 20

SLOT MACHINE

Questo gioco fa sì che il VIC funzioni come una slot machine. Il vostro gruzzolo di partenza è di 80 pezzi, dei quali ne potete giocare fino a 5 per volta. Le combinazioni possibili sono tre su piani orizzontali e due su piani diagonali e le prime hanno ordine di precedenza sulle seconde. Le combinazioni vincenti sono visibili premendo il tasto *V*. Per far partire il gioco premere il tasto *f1*. Il tasto *C* vi permette di puntare e il tasto *P* fa partire la slot machine. Il numero visualizzato nel quadrato blu in campo inverso è il numero di pezzi che avete a disposizione e si aggiorna automaticamente dopo ogni puntata ed in caso di vincita.

Cartuccia TF/9300-18
L. 37.000 IVA esclusa
VIC 20

THE ALIEN

Immedesimatevi nei panni dell'alieno, siete all'interno di un labirinto e avete speranza di sopravvivere solo catturando gli esseri che sono all'interno del labirinto. Per far ciò avete la possibilità di piazzare trappole, aspettando poi che vi cadano dentro gli esseri presenti nel labirinto; avete 5 tentativi a vostra disposizione. È importante che facciate in fretta a sgonfiare la trappola una volta che vi è caduto dentro l'essere, perchè più tempo aspettate più quest'ultimo diventa pericoloso ed è facile che vi distrugga. Ogni volta che uccidete un essere il vostro punteggio aumenta di un numero che varia fra 100 e 1000. Ad ogni labirinto ripulito vi viene assegnata una bandierina. È possibile registrare i punteggi più alti raggiunti insieme al nome del giocatore. Le istruzioni vi vengono fornite dal programma.

Cassetta TF/9406-00
L. 60.000 IVA esclusa
VIC 20 con espansione 3 Kbyte RAM

Programmare, studiare, forse sognare... (parte II)

Questa è la seconda puntata della risposta al "perito industriale capotecnico in Informatica" Roberto Bertucci. Questo napoletano verace, ricordo velocemente a chi non avesse letto la prima parte (pubblicata sul n. 35 di **Bit***), mi chiedeva di sciogliergli il dilemma: mi iscrivo all'Università o mi metto a lavorare? Alquanto pilatescamente, ma anche spinto a ciò dal duplice motivo dello spazio ridotto e dell'esser detta missiva pervenuta, ahimé, fuori termine per un responso valido, ho preferito rimandare proprio il discorso sulla validità delle nostre attuali facoltà di Scienze dell'Informazione.*
Visto che, a tutt'oggi almeno, nessuna lettera autoesplodente mi è pervenuta (sintomo di consenso, indifferenza o di ... ringhiosa attesa di ulteriori compromissioni da parte mia?) mi riallaccio anzitutto a quanto detto al termine della scorsa puntata. Ossia che, se è vero che oggi troppo abissale e cronico è il ritardo della scuola in genere rispetto alla realtà della professione, è parimenti vero che ormai è quasi utopistica ogni pretesa di diplomarsi bell'e pronti per entrare, senza scosse né aggiustamenti di sorta, nel mondo del lavoro. Esso è infatti così mutevole e vario che di figure professionali ne crea e distrugge di continuo. Accennavo anche al fatto che, mentre il mestiere di "informatico" sta diventando merce ogni giorno più rara, cresce invece la necessità che nozioni generali e spunti spiccioli d'informatica vengano possedute da "professional" (come dicono gli anglofoni, intendendo manager e quadri di buon livello...) d'ogni tipo. Se questa tesi è vera (e non son certo isolato nel sostenerla) a me sembra che essa rafforzi il ruolo, preciso, che una facoltà di Scienza dell'Informazione deve in tale contesto assumere: quello cioè di fucina per la qualificazione ad alto livello degli informatici puri. *Costoro dovranno, una volta laureati, dedicarsi o alla ricerca oppure all'esercizio di una professione rivolta alla creazione e sviluppo di software avanzato, compresi gli applicativi, però impostati in modo, ad esempio, sufficientemente flessibile e parametrico da consentire agevolmente tutti gli adattamenti e "personalizzazioni" mantenendo le economie di scala che, ormai, solo i prodotti generalizzati permettono. Un siffatto dottore in Informatica, voglio quindi precisare, sarà particolarmente "bravo" nell'impostare razionalmente tali prodotti (secondo cioè i principi di efficienza e logicità che solo l'Informatica "pura" possiede), però deve anche rassegnarsi all'idea che, sul terreno più spicciolo, saranno sempre più bravi di lui, salvo eccezioni, gli ingegneri e i ragionieri: ciascuno, nel suo ramo, è unico nel conoscerne le esigenze e farne l'analisi come si deve. A questo punto dovrebbe essere assai chiaro che personalmente - anche a rischio di accelerare l'inoltro di quelle lettere al plastico sopra paventate - sconsiglio decisamente i periti capotecnici di darsi alla Scienza dell'Informazione. E questo (salvo eccezioni ben s'intende) vale anche (o forse ancor più) per i capotecnici-informatici: la loro preparazione è infatti, tutto sommato, eminentemente pratica e in parecchi casi rischia di smarrirsi nei bizantinismi sottili dei task concorrenti, degli interpreti metalinguistici o metasemantici, nelle bizzarrie dei procedimenti euristici e stocastici e via sofisticando. Implicita nei discorsi precedenti era, palesemente, l'idea che le nostre facoltà di Scienza dell'Informazione possiedono per lo più questo carattere prevalentemente astratto. Significa anche che approvo tale impostazione? Si e no. Si perché l'astrazione, checché ne pensino i praticoni, è pur sempre il grado più alto della scienza occidentale. No se ciò comporta l'avallo dei malvezzi più atroci, a partire da quello per cui, in molte, troppe di tali italiche facoltà, si programma pochissimo, il che le fa rassomigliare a facoltà di Fisica senza esercitazioni di laboratorio. Questo serio difetto è peraltro anche legato al sovraffollamento della nostra Università in genere e che, in questi momenti d'informatica trionfante, ho l'impressione che coinvolga queste "nuove" facoltà (che tali invece non sono, essendo state istituite ormai da molti anni...) quasi quanto quelle di Lettere e Giurisprudenza.*
Sintomatico il caso avvenuto in una di queste (quale? non ve lo dirò mai): la decisione, di per sé encomiabile, di acquistare un certo numero di microsistemi ha accresciuto le frustrazioni degli studenti (provocando persino qualche fuga), perché sono esplose le contraddizioni dell'inadeguatezza delle strutture in questa tragica università degli assenti!
Conclusioni? Sarò qualunquista (ma che altro si può fare, in attesa di Riforme coraggiose quanto mitiche?), invitando tutti coloro che, malgrado tutto, siano allettati dall'idea di darsi all'Informatica astratta, ad iscriversi pure. Però oggi vale l'aiutati-che-dio-t'aiuta, il che, in termini spiccioli (e, giuro, senza che tangenti di sorta me ne vengano in tasca) si coniuga così: almeno compratevi un personal.

Programma Grand Prix

Egregio signor Barzaghi, la prego di scusarmi se le scrivo e se può dedicarmi un po' del suo tempo. Ho un computer della Commodore "PET 2001 a 32 Kbyte"; sfogliando **Bit** n. 27 (Aprile '82) ho trovato Grand Prix, le faccio i miei complimenti per tutti i suoi programmi.
Purtroppo, dopo averlo mandato in RUN si ferma alla linea 13008 ed i seguenti errori:

1) Nel riordino della griglia di partenza nelle prime 4 file il tempo delle prove è 0'00"000 e dalla 7ª alla 12ª mi segna 1'10"037.
2 Alla partenza, quando deve dare il miglior tempo sul giro (solo il 1°/2°/3°/6° pilota mi segna solo i centesimi) quando poi arriva all'ottavo pilota (Reutemann) mi segnala DIVISION BY ZERO ERROR IN 13008; certe volte anche in 367 e 1650.

Invece per il tempo di perdita o di sosta ai box lo segna tutto giusto, esempio: 2'32"145. Le assicuro che dopo aver scritto il programma e trovato gli errori, l'ho ricontrollato tutto per ben due volte.
Desidero sapere se ci sono degli errori di copiatura del giornale, oppure qualcos'altro. La prego di mandarmi un listato del programma, così lo riguarderò per la terza volta e mi dica lei il suo parere.

Gianni Favazzi

Rispondendo al signor Favazzi, vorrei indirettamente rispondere a tutti quei lettori che ci hanno inviato richieste di chiarimenti o di correzioni ad articoli e programmi tratti dalla rivista. Se è già difficile diagnosticare degli errori avendo sott'occhio un programma che si è digitato personalmente, è intuibile la difficoltà che si trova ad affrontare chi, come il sottoscritto, viene invitato a dedurre possibili errori in base alle scarse informazioni contenute nelle lettere dei lettori.
Per quanto riguarda il caso specifico del signor Favazzi, lo inviterei a rivedere sul n. 27 di **Bit** *i REMarks al programma in questione, soprattutto per quanto riguarda il formato delle frasi DATA per i piloti iscritti. Non facendo lei riferimento ad un particolare Gran Premio, posso solo fare delle ipotesi al riguardo; le ricordo comunque che i piloti che non hanno preso parte ad una gara devono essere accoppiati ad un tempo in prova fittizio che li escluda dallo schieramento di partenza (per esempio 9'99"999).*
In passato, molti errori che ci sono stati segnalati non erano altro che banali errori di ricopiatura, dovuti anche alla scarsa leggibilità dei listati. La Redazione, comunque, non può farsi carico dell'invio di listati o di risposte ad personam a problemi vari, poiché probabilmente non ci resterebbe il tempo di sviluppare altri nuovi programmi.

Dove sono i Sinclair Club?

Sono uno studente in informatica, felice possessore di un ZX81, desideroso di scambiare il mio software con altri ragazzi della mia età. Purtroppo abitando in un centro isolato mi è impossibile mettermi in contatto con altri utilizzatori di Sinclair.
Come posso fare?

Gianni Galimberti - Tavarone (SP)

Niente di più facile! Su queste stesse pagine, nella rubrica Microflash, *si parla del* Sinclub, *un club nazionale che raggruppa gli utenti di Sinclair.*
In seguito verranno date ulteriori informazioni, nel frattempo spedisca il coupon presente nella rivista.

Di Bella risponde: Grand Prix si gioca con queste varianti

Con riferimento all'articolo "*Grand Prix*" di G. Di Bella apparso sul numero 33 Novembre '82, vorrei chiedere le variazioni da apportare al programma perché si possa comandare il movimento con i tasti S e D come indicato alle linee 120 e 130. Sul mio 4032 non hanno nessun effetto.

Battista Piaggi - Torino

Precisa l'autore: "occorre sostituire nella riga 2010, al posto di PEEK (151) = 40, l'istruzione PEEK(151) = 83; invece nella riga 2020 la PEEK(151) <> 47 va riscritta così: PEEK(151) <> 68". E qui Di Bella ringrazia commosso per l'attenzione prestata al suo game. Non c'è di che.

Feedback

Una voce stonata? Non direi

L'intervento sul numero 37 che **Bit** ha dedicato alla didattica con il computer (di cui, già che ci sono, intendo vivamente complimentarmi) della signora Fasano Petroni mi è parso l'unico o quasi in stridente contrasto con tutto il resto. Il suo "diverso parere" sarà anche anticonformista e controcorrente e quindi in linea con quello che io chiamo ... conformismo dell'anticonformismo, oggi così di moda. Però non mi trova quasi in niente d'accordo. L'autrice è naturalmente libera di esprimere l'opinione che le sembra, però ha anche il dovere di informarsi meglio e allora scoprirà che i personal computer in molti casi stanno entrando nella scuola in modo massiccio. La famigerata disposizione da lei citata che sottopone a preventiva autorizzazione ministeriale l'acquisto di apparecchiature informatiche d'ogni genere (si noti: "d'ogni genere" comprende anche sistemi di sviluppo e qualunque strumento comprendente microprocessori, il che - preso alla lettera - includerebbe la totalità della moderna strumentazione elettronica!) viola l'autonomia amministrativa dei Consigli d'Istituto che pertanto, non appena possono e si rendono conto di ciò, provvedono sottobanco ma legittimamente a fare acquisti con fondi propri, infischiandosene del centro di potere costituito dalla commissione romana ed approfittando dei bassi costi che oggi hanno i personal e della possibilità che molti forniscono di graduare nel tempo l'investimento: si inizia con diverse unità centrali ed unità video più qualche floppy drive ed una sola stampante, in seguito ci si espande. L'autrice invece continua a ragionare come ai tempi in cui un microcomputer con 16 Kbyte di RAM si pagava decine di milioni (se invece intendeva stigmatizzare un perdurante malcostume, analogo, benché su scala ridotta, a quello di Torino, in cui si è pagato 8 miliardi un computer che costava 800 milioni doveva dirlo più esplicitamente: mi avrebbe trovato consenziente). Quanto alle sue idee sull'informatica povera e dei fogli di programmazione (*veramente l'autrice ha parlato di flow chart* NdR) è chiaro a tutti che esse appartengono al passato, quello in cui - come ricorda il prof. Tonti sullo stesso numero speciale - le macchine si alimentavano a schede ed i programmatori nella sala sacra neanche ci mettevano piede, lavorando a tavolino. A quei tempi i corsi di programmazione si tenevano *senza* alcun impiego o quasi di macchine e gli allievi scalpitavano giustamente.

In conclusione non vorrei sbagliare ma mi sembra di ravvisare in simili posizioni il timore - comune a parecchi insegnanti - di venir scavalcati dagli allievi che, messi davanti al computer, liberamente apprendano più di quanto i loro insegnanti sappiano. Anche l'accenno alla "mancanza di assistenti" denuncia la mentalità di chi vuol vedersi servito da schiavetti e rifiuta di sporcarsi le mani con strumenti che, invece, non hanno nulla di misterioso (e sono facili da aggiustare, perlomeno nella manutenzione ordinaria).

(lettera firmata)

La precedente lettera è frutto di un merge tra più d'una, talora dai toni ancora più accesi di quanto non traspaia dalla ricucitura che abbiamo operato, cercando di rimanere sostanzialmente fedeli al pensiero degli autori. Questa violenza verbale, mi si lasci anzitutto osservare, dimostra che il "diverso parere" era davvero stimolante. Quindi ben ha fatto Cavalcoli a proporne l'inserimento nello speciale. Un'altra osservazione che, senza erigermi a difensore d'ufficio, mi sembra doveroso fare è che molte delle cose dette rassomigliano troppo deplorevolmente alla gratuità dei processi alle intenzioni oppure sono frutto di un fraintendimento del pensiero dell'autrice. Costei ha invece il merito di aver sollevato problemi reali e tra questi è assolutamente necessario - come ci sembra di aver detto nel "manifesto sul personal computer nella scuola" - tener conto senza alcuno spirito terroristico della situazione attuale. Questa vede i seri problemi della formazione dei docenti da un lato e, dall'altro, della proliferazione "selvaggia" delle macchine con tutti i rischi (in atto già da tempo: qui do ragione ai vari contestatori sopra riportati in cooperativa, anche se il numero di PC nelle scuole, superiore a quello che traspare in superficie, non è ancora così ampio) di una loro sottoutilizzazione, male peraltro oscuro e cronico nella nostra scuola. Senza volermi pronunciare sulla legittimità giuridica-politica della pianificazione dall'alto di questo fenomeno incontrollato - o incontrollabile? si pensi che qualcosa di analogo sta accadendo anche nelle aziende comprese quelle private: il tutto si ricollega all'eterna diatriba tra autonomia e controllo tra decentramento e centralizzazione - a me pare infine che il parere della gentile docente risulti ispirato a grande equilibrio e, insieme, degno di essere meditato. Ribadiamolo: l'autrice non si è dichiarata nemica del personal, ma solo dubbiosa circa i pericoli della loro invadenza totalizzante. Insomma si tratta di trovare una giusta mediana tra gli opposti estremisti della teoria pura e del puro empirismo, del batch e dell'interattività immediata e ultracontinua. Esemplificando, a me sembra che oggi vi sia una certa esagerazione nella mentalità di chi crede solo al programma impostato e subito verificato, al punto che, dinnanzi alla più banale coppia di righe di codice, non si pronuncia se prima non le ha digitate al terminale! Di questo passo 2 + 2 fa 4 solo se ce lo dice la macchina e così ci scordiamo che, alle sacrosante esigenze della verifica sperimentale (specie con linguaggi pasticcioni come il BASIC, non mi stancherò di dirlo, con tutte le loro ambiguità "semantiche"...), occorre associare la riflessione che quasi tutto (compresa la moderna tecnologia) nase da idee partorite nella nostra testa. Su un piano tecnico poi è dimostrato che programmi direttamente sviluppati al terminale risultano più rapidi da produrre ma in genere assai meno chiari e per nulla documentati. I guru della moderna informatica, come Hoare e Dijkstra, concordano nel raccomandare studi preliminari sulla carta, con flow o diagrammi di struttura. Un buon softwarista, sostiene il primo dei due, sviluppa lentamente il suo lavoro che però - meditato e sviscerato a dovere - gira al primo colpo. E, nel seguito, è più facile da comprendere nella sua struttura e suscettibile di essere modificato senza problemi. Insomma è soprattutto una questione di buon senso.

■ ■ ■

Peek, Poke, qui-pro-quo e sarchiaponi

Forse non ci siamo capiti bene. Io davo per scontato il significato delle istruzioni PEEK, POKE e CALL; non desideravo assolutamente

vedere la spiegazione di una pagina e mezzo (non dico certo che è poco) del suo rispettabilissimo commento da buon maestro (*ringrazio commosso!* G.G.).
Quello che cercavo di capire è cosa possono creare queste benedette istruzioni cercando di fare degli esempi con piccoli programmi (vedi l'*Asta che cade*). L'articolo sul numero 35 **Bit**, sull'utilizzo dell'Apple come controller di 8 canali audio, ne è una delle tante testimonianze.
Non è luogo migliore per questi programmi una bella rivista di elettronica applicata? Non sono sciupati in **Bit**?
Vede, Signor Giaccaglini, gran parte delle riviste italiane e non, cercano di spiegare quel famoso "top secret" della mia precedente. Lei, a quanto pare, da già per scontato tutto questo, io no!
Mi sembra insomma che la rivista sia abbastanza sofisticata e che l'accesso sia per l'élite di ingegneri e fisici.
Eppoi ancora: non è che ci sia stato un abbinamento tra Brunazzi e patitazzi? Se la ricorda la questione in tribunale per Roberto Benigni su "Woitylaccio" (si scrive così)?
P.S. Ma è lecito tagliare le lettere pubblicate?

Danilo Brunazzi - Ponte Taro

Le lettere si tagliano e sintetizzano per ragioni di spazio oppure di censura in senso "benevolo". Mi spiego: qui, ad esempio, ho tolto i salamelecchi, lasciando gli improperi. Questi, beninteso sono sacrosanti. Però occorre precisare che, accanto ai begin-the-beguine, ovverosia coloro che nella discoteca dell'informatica non sanno ballare nemmeno la danza del trescone (che, detto tra parentesi, non so neanch'io cosa sia...) ci sono anche i saputelli, ai quali articoli su argomenti avanzati come quello incriminato interessano eccome. Accontentare tutti diventa così piuttosto difficile, ne convengano anche i Brunazzi-patitazzi (la cui freddura, francamente, non ho capito, anche se "sento" che deve essere tremenda, tale da porre definitivamente in ombra qualsiasi boutade, calembour e jeu de mots passati e futuri...). Oltretutto far felici i principianti è parimenti difficile: infatti si scopre quasi regolarmente (e regolarmente qui cascano gli asini come il sottoscritto) che essi sanno già tutto o quasi sull'argomento, solo che vorrebbero degli esempi specifici. In casi del genere il secondo round ha pur'esso un copione quasi fisso: si forniscono i casi e casini richiesti, al che il principiante-però-mica-tanto replica seccato: ma io queste cose le sapevo già... Ad essere cattivi verrebbe la tentazione di avanzare l'altroce sospetto che qualcuno (ed i presenti sono, per educazione, sempre esclusi!) i concetti li conosca ma non li sappia applicare troppo bene. Di fronte a tale ipotesi le possibilità sono due: o suggerire di esercitarsi, esercitarsi, esercitarsi oppure ... di copiare, copiare, copiare. Nulla di scandaloso infatti: in Informatica questo principio è stato autorevolmente teorizzato dai massimi santoni come Hoare, Dijkstra ecc.
Qui però è il momento di piantarla con (l'apparente) castigazione del malcapitato Brunazzi che, sutto sommato, ha ragione. In effetti l'EDP abbonda come non mai di "sarchiaponi" innumerevoli, bestie che cioè, come già Walter Chiari ed io stesso diverse volte ho avuto modo di dire, tutti fingono di conoscere. Se l'impegno a migliorare la situazione è (prevengo senz'altro la giusta rilevazione dei vari Brunazzi) superfluo e risibile, se la neonata rubrica "Il Ricettario" è un palliativo (spero sono non divenga... "Il Rigettario"!), se il dire che anche le altre riviste concorrenti sono, almeno per la statistica, altrettanto zeppe di sarchiaponi suscita solo spallucce e derisioni ... ebbene no, non darò le dimissioni, non mi dedicherò al giardinaggio o al windsurf. Persevererò, novello Sisifo della microinformatica popolare-ma-non-troppo, ad aggiungere parentesi con delle NdR (Note della Redazione) ai passi più oscuri dei nostri sempre più saputi collaboratori vicini e lontani, a supplicarli con sempre più umili fiati di essere chiari, e poi ancora chiari e, infine, a pregare pubblicamente costoro di affrontare argomenti veramente divulgativi e, al tempo, non banali. Un giorno, chi sa, qualcuno mi dedicherà una lapide.

Le schede sacre

In una sua precedente risposta su *Feedback* lei ha parlato di "vacche sacre" che ancora si adorano nelle nostre scuole, di vario ordine e grado, che tengono corsi o addirittura rilasciano diplomi o lauree in Scienza dell'Informazione. Senza voler generalizzare, le confermo che quel che dice è vero: ancora oggi, in uno dei nostri più prestigiosi atenei, gli studenti o i laureandi che svolgono esercitazioni di programmazione debbono far la coda col loro pacchettino di schede perforate da 80 colonne (le stesse, credo, inventate nell'Ottocento). Non le sembra un tantino anacronistico e, oltretutto, alquanto costoso?

Gerolamo Sandrucci - Cesena

È difficile, sulla base di indicazioni tutto sommato così generiche del nostro lettore, entrare nel merito delle scelte operate dai docenti: vi possono entrare in ballo problemi di ammortamento di impianti costosi che oggi magari sarebbe un delitto riacquistare ex novo quando esistono da tempo mini e microcomputer enormemente più convenienti per la didattica ma che fino a qualche anno fa sembravano a molti insostituibili. È pure ipotizzabile, visto che si sta parlando anche di università, che grandi e grandissimi sistemi siano tutt'ora insostituibili per compiere ricerche di tipo avanzato e che, non senza valide giustificazioni, il concedere a quei servi della gleba che nella piramide accademica sono gli studenti (sopra c'è il Rettore-imperatore, sotto i vassalli-cattedratici, seguono i valvassini, valvassori, valvassetti degli assistenti e tirapiedi vari...) microsecondi del preziosissimo tempo macchina di simili bestioni sia tutto sommato un atto di regale generosità. Detto di passaggio, l'accesso al terminale è senz'altro giusto che venga inibito alle sterminate masse dei proletari: lo spreco del tempo macchina sarebbe d'una diseconomia riprovevole, così costringere un discente al paziente esercizio di autoperforarsi le card inventate da Hollerith (attorno al 1880, per risolvere problemi del censimento USA, credo) può persino non essere un male: è un'abitudine all'umiltà e, insieme, alla massima precisione e meticolosità d'impostazione, specie per chi vuole evitare nuove estenuanti code in un turno successivo. Però la tentazione di far due conti viene lo stesso. Vediamo: 600 schede (tale è il contenuto d'una confezione) contengono 600x80 = 40.000 caratteri e costano attorno alle 5 mila lire. Supponiamo, per semplicità, che altrettanto suppergiù si paghi un floppy disk, in cui però la capienza minima è più che doppia. Inoltre il tempo macchina, coi personal, è valutabile in termini di consumi d'energia elettrica pari a quelli di un televisore più il mediocre ammortamento di un costo che non supera i cinque milioni, stampante inclusa. E non si venga a dire che in simili "giocattoli" si lavora soltanto con oggetti soft volgarucci come BASIC e CP/M, perché ormai lo sanno tutti che già viene offerto il Pascal e, a spender qualcosina in più, quel sistema operativo più ... Unix che raro. Oggetto - sempre in base agli echi che me ne giungono - di un nuovo culto esoterico presso piccole élite d'informatrici paludatissimi, anche se di giovanil pelo, adesso anche questo sacro mostro sta invadendo impudentemente il libero mercato. Signori dell'accademia che aspettiamo a darlo in pasto anche alle "vostre" fameliche masse?.

Errorucci nelle Torri in Pascal: grazie

Nel listato 1 "*Soluzione in Pascal del gioco delle Torri di Hanoi*" dell'articolo "*Far ricorso alla ricorsività*" di L. Fonti apparso a pag. 71 del numero 28 di **Bit** vi segnalo alcuni errori formali. Deve essere IF (K=1) ... e non IF(N=1)... perché la stampa deve ovviamente essere fatta per ogni "discesa" della variabile K a uno. Inoltre nell'IF il ; prima dell'ELSE va eliminato perché il punto e virgola, in Pascal, è un separatore di istruzioni e quindi in particolare non può essere posto all'interno dello statement IF ... THEN ... ELSE a meno che non sia composta da più istruzioni la parte compresa tra THEN e ELSE. E mentre in un programma Pascal vi possono essere più END che BEGIN (si pensi ad esempio al CASE o alla descrizione di record) l'opposto è impossibile, pertanto va eliminato il primo BEGIN del listato.

Gennaro Ragucci - Napoli

Giuro che di due tra gli errori segnalati: il BEGIN iniziale di troppo e l'uso di variabile globale N in luogo di quella parametrica K, m'ero accorto anch'io, ma poi la mia direttiva di stampare un listatino sostitutivo non è stata evasa (o me ne sono dimenticato? Chi si ricorda...). Grazie anche a nome dell'autore, che con il "classico" programmino voleva solo dare un esempio e quindi si è preoccupato unicamente della sostanza logico-algoritmica.

Feedback

Mentalità informatica

Trovo piuttosto interessanti articoli come quelli di Albanesi (*"Golbach risolto more informatico"*, **Bit** n. 26 pag. 101 o il più recente "*Il computer: una struttura di calcolo con funzioni decisionali*", n. 35 pag. 45) perché mettono tra l'altro in rilievo una nuova mentalità, rispetto a quella imperante sul tradizionale terreno della matematica. I problemi però mi sembrano aperti e non tutti privi di risvolti critici o, addirittura, discutibili. Non mi riesce di fare (e, sicuramente, mi manca la necessaria competenza) esempi significativi al riguardo, tuttavia ritengo che una eccessiva disinvoltura e mancanza di rigore siano gli scotti che si rischia di pagare. Sarei contento che l'argomento possa essere dibattuto sulla vostra rivista di cui, malgrado alti e bassi, continua ad apprezzare lo spirito vivace e stimolante.

Giovanni Corbetta - Asti

Neanch'io, specie così, su due piedi, riesco ad escogitare casi notevoli e soprattutto significativi. Comunque, in attesa di eventuali interventi qualificati, proverò a svolgere un poco estemporanee considerazioni. Anzitutto va detto che l'Informatica da diverso tempo è stata capace di assurgere a ruoli e dignità di scienza con la "s" maiuscola, trasformandosi da tecnica semi-empirica a teoria sistematica e, perciò, rigorosa quanto occorre e basta. Grossi studiosi hanno così delineato organiche teorie sugli algoritmi, sui processi concorrenti con tutte le loro diavolerie di "semafori", primitive ecc., sulla ricorsività. Queste a loro volta hanno contribuito a definire nuovi linguaggi di programmazione, più o meno vicini ai criteri delineati nell'informatica di punta e nel settore dell'intelligenza più o meno artificiale. Il problema aperto in qualche modo dal lettore può allora per sommi capi, articolarsi su due punti:

a) quanto di tutte queste belle cose sia da considerare teorico o, invece applicativo;
b) quanto la nuova Scienza dell'Informazione debba essere tributaria della Matematica e in che misura, invece, se ne distacchi.

Il secondo punto è probabilmente il più cruciale. Non solo infatti è avvenuto ciò che è già regolarmente capitato in altre discipline originariamente nate come tecniche (si pensi a Leibniz e Newton che, nell'inventare limiti e derivate, si preoccuparono solo di fabbricarsi degli strumenti adeguati alla nuova meccanica, lasciando ai posteri-matematici i non pochi problemi della loro definizione rigorosa) ma, almeno così mi sembra si deduca tra l'altro dagli articoli citati, l'informatica pretende di superare per molti versi le pastoie che il rigorismo analitico pongono sul terreno applicativo. Che da qui possano derivare taluni "pasticci", per il fatto stesso che c'è una qualche contraddizione nella stessa affermazione di voler "rigorosamente impostare processi non-rigorosi" (nobilitando la cosiddetta "fuzzy logic" ad esempio) è cosa che ciascuno intuisce da sé.

Quel che è poco ma sicuro è che informatici e matematici operano con criteri sovente divergenti. Provo a illustrarlo con un banalissimo ma buffo aneddoto. Un programmatore ha bisogno di una routine per calcolare la sommatoria degli interi da 1 a N. Senza pensarci troppo scrive queste righe di BASIC:

```
1000 T = 0
1010 FOR I = 1 TO N
1020 T = T + I
1030 NEXT I
```

Subito un matematico suo amico lo deride, rinfrescandogli una semplicissima formula, che, per suprema ironia della sorte, riduce anche spazi e tempi elaborativi e mnemonici:

```
1000 T = N★(N+1)/2 (!)
```

Il punto esclamativo sulla destra è aggiunto a somiglianza di quanto si fa nei resoconti scacchistici...

La morale di questa favoletta la lascio a chi legge, non senza accennare al fatto che informatici "fanatici" fanno ricorso-a-ricorsività e ad alberi in modo talora eccessivo, trascurando possibili alternative più sintetiche e veloci. La versione informatica si presenta però, anche in simili casi, più immediata, spontanea e "naturale" (come nell'esempietto sopra in cui all'elaboratore si affida il compito di ... contare sulle dita) e risulta poi insostituibile in tutti quei casi in cui la matematica "classica" non ci fornisce formule risolutive precostituite. Insomma l'EDP ci offre strade che portano a Roma, anche se ignoriamo (e non esistono) scorciatoie.

Storchi & Torri e Mantovani: se ci siete, battete un colpo

Diversi lettori ci hanno segnalato malfunzionamenti dei programmi per il VIC 20 "*The Wall*" comparso sul n. 33 di **Bit** e CREO, LISTA, STAMPA di Mantovani. In particolare Bruno Montresor (via V. Locchi 25 - Verona) osserva che "sono cose che scombussolano" aggiungendo che, nonostante tutto resterà nostro lettore. Commossi da tanta fedeltà ci siamo rivolti alla GBC dove ci hanno informato che, avendo anch'essi riscontrato simili "bachi" hanno eliminato dal catalogo quel prodotto e l'hanno rifatto. Ora è disponibile nel catalogo Softbank della Rebit col titolo CREO/LIST e il codice TF/9350-02. Quanto a noi, ahimé, abbiamo perso l'indirizzo dell'autore. E, quel che è peggio, abbiamo perduto pure quello di Storchi e Torri! Un lettore dalla firma illeggibile ci segnala, a tale proposito, le inesattezze ed oscurità seguenti:

- i caratteri speciali sono su un fondo bianco e si confondono coi normali;
- alla riga 130 manca un'istruzione, come pure gli apici;
- per chi non ha joystick si dovrebbe specificare cosa mettere nelle righe 250, 300, 310, 320 e 330;
- alla riga 730 si invia ad una linea inesistente
- alla riga 1030 dovrebbe esserci IF FD, invece manca una "F"...

E il tutto non basta a far girare il programma. Dopo un pubblico nostra culpa (per quel che potrebbe eventualmente riguardarci) invitiamo i tre autori a farsi vivi con opportuni errata corrige. Adesso comunque le cose miglioreranno nettamente con la nascita di **Bit-RP** e le molte macchine su cui sistematicamente i programmi (che verranno accettati solo se su cassetta) saranno testati. Abbiamo anche diversi VIC 20.

PICCOLI ANNUNCI

La rubrica è aperta a tutti i lettori. Le inserzioni sono completamente gratuite. I testi degli annunci devono essere inviati utilizzando l'apposito tagliando riportato in fondo alla rivista. Chi volesse evidenziare il proprio annuncio, con il tratteggio (■■■■) ai lati, deve unire l'importo di L. 5.000 anche in francobolli.

LO SPAZIO DEL RIVENDITORE

I rivenditori possono inserire i loro annunci all'interno di questa rubrica a condizioni particolari. Le tariffe e gli spazi disponibili sono forniti dalla Concessionaria di pubblicità Reina s.r.l. - Via Washington, 50 - 20149 Milano - Tel. (02) 4988066 - 4988067 - 4988068 - 4988069 dietro semplice richiesta, anche telefonica.

BORSA

Hardware

Vendo 300Klire **ZX80** trasformato ZX81 reverse video + 2 libri inglesi con 100 programmi circa + espansione 4K completa integrati + 16K (necessario per quest'ultima cambiare trasformatore) elegante contenitore plexiglass giallo con "vera tastiera" + tasto reset + cassette regalo e manuale nuova ROM (regalo anche la vecchia).
Sabbatini Paolo - V.le San Marco, 49 - 30173 Mestre (VE) - Tel. 041/974029

Vendo **ZX81** con 16K RAM, completo di accessori e corredato di circa 300 programmi, tutto a L. 380.000.
Vendo i programmi anche separatamente a L. 35.000 in blocco.
Bugli Sauro - Via Delle Ortensie, 2 - 50142 Firenze - Tel. 055/701103

Vendo **PET 2001** completo di ventilatore e sonoro vendo a 700.000, completo di manuale e 20 programmi vari con 10 nastri omaggio.
Bastianelli Giuliano - Via Gioberti - 71030 Foggia - Tel. 0881/22805

Vendo stampante **Epson TX-80B** nuova compatibile con commodore - Apple VIC.
Bellù Dino - Via Ballerini, 26 - 20038 Seregno - Tel. 239092

Vendo **HP41-CV** con lettore di schede, stampante tipo 82143A, moduli per memory e per function più schede e programmi per programmazione sintetica, lo HP41-CV è stato acquistato nel maggio 1982, prezzo/i da definire.
Antonietti Claudio - V.le Faenza, 1 - 20142 Milano - Tel. 810687

Compro **stampante 3022** opp. 4022 per C.B.M. Commodore 3032 - solo in zona e se in buone condizioni.
Albani Alvaro - Via Castelfidardo, 7 - 47037 Rimini - Tel. 0541/25765-0541/52335

Vendo Computer **Personal N.E.** video tastiera 8K RAM reg. cassette mobile carrello 6 cassette programmi valore kit 1.500.000 vendo per urgente realizzo a L. 1.100.000 occasione unica e irripetibile per iniziarsi all'informatica.
Cosolo Eugenio - Via Marconi, 19 - 34070 Turriaco (GO) - Tel. 0481/769474

Vendo computer **Zenith Z89** CPU Z80 memoria 64K 1 floppy disk sistema operativo CPM2.2 linguaggio Basic 80 micro-soft completo di tutti i manuali usato solo per apprendimento come nuovo L. 200.000. - telefonare ore serali.
Grazia Giuseppe - Via A. Saffi, 49 - 40131 Bologna - Tel. 051/557839

Vendo **Apple II** + language card + monitor 12" + drive + scheda colore PAL + interfaccia parallela L. 3.000.000 in omaggio visicalc.
Manuli Alberto - Via Donizetti, 23 - 20122 Milano - Tel. 797877

Sharp MZ-80K con Basic doppia precisione, manuali, interfaccia per centronics L. 1.600.000 - inoltre scheda Z80 microsoft con CPM per Apple II sconto 20% - telefonare ore pasti.
Pieri Luigi - P.zza della Torretta, 20 - 00186 Roma - Tel. 06/6782183

Vendo causa partenza militare **ZX80** sinclair 1 K RAM completo di manuali cavi e 30 programmi scritti ROM 8K + ROM 4K, registratore a cassette il tutto (8 mesi di vita e usato pochissimo) a L. 250.000 intrattabili - telefonare ore serali.
Lupo Fabio - Via 2 Dicembre - 16157 Genova - Tel. 632008

Vendo **ZX80** Sinclair N ROM 8K + 3 K RAM con alimentatore + manuali a L. 150.000.
Magnanini Mauro - Via Frutteti, 123 - 44100 Ferrara - Tel. 0532/21893

Vendo **Sharp MZ80B** come nuovo completo di espansione RAM 64K e di extension unit - 6 posti scheda - a L. 2.800.000 - in omaggio interfaccia parallela centronics autocostruita per stampanti tipo Epson.
Martelli Giorgio - Via Galeotti, 22 - 40127 Bologna - Tel. 501966

Vendo **VIC20** + super expander + combinatore telefonico + sintesi vocale + programmi su cassette + joystick vendo a L. 700.000 o cambio con scheda lemon 2 telefonare ora di cena a: Fraccaro Lorenzo - Via Capparozzo, 35 - 36100 Vicenza - Tel. 505828

Vendo **stampante PC100C** + SSS math, statistica, elettronica, completi di manuali - L. 300.000 - telefonare ore serali a Ugo.
Ribaga Ugo - Via J.Da Tradate, 7 - 20155 Milano - Tel. 02/361521

Vendo computer **VIC-20** un mese di vita al miglior offerente. Regalo interfaccia per registratore, televisore, cassetta con molti giochi e programmi.
Codognola Andrea - Via Morandi 2/C - 20097 S. Donato Milanese (MI) - Tel. 02/5271481

Vendo **TI-59** ottimo stato - vari programmi su schede - schede vergini - modulo base - modulo RPN simulator - alimentatore - relativi - manuali - L. 250.000. - Possibilmente acquirente zona Bologna e provincia.
Braga Giuseppe - Via Bellaria, 28 - 40139 Bologna - Tel. 545469

Vendo **TRS 80 mod. 1** livello 2 - 16K RAM - completo di registratore a cassette, alimentatore, monitor - prezzo interessantissimo.
Codato Davide - Via B. Galuppi, 11 - 30171 Mestre (VE) - Tel. 041/962443

Vendo **CBM 2001** PET 8 Kbyte nuove ROM unità a cassette e video a fosfori verdi - in ottimo stato con manuali in italiano L. 700.000 trattabili.
Nardello Gianfranco - Via Mazzini, 98 - 21050 Bisuschio (Varese) - Tel. 0332/471583

Vendo **Micro N.E.** completo di LX380/1/2/3/4/5/6/7/8 + LX392 in elegante contenitore, il tutto perfettamente funzionante e usato pochissimo - scrivere o telefonare ore pasti a:
Mingardi Carlo - Via F. Guardi, 5 - 40133 Bologna - Tel. 051/313266

Vendo **Atari 800** 48K + 410p + paddle + joystik music composer + star raider vendo L. 2.000.000 o cambio con apple 48K con modulatore tv tutto imballato con tre mesi di scarso utilizzo.
Battelli Maurizio - Via Roma, 100 - 61013 Mercatino Conca (Pesaro) - Tel. 0541/970148

Vendo **CBM 4032** + stampante epson MX-80 + doppio floppy computhink (200 + 200 K bytes) + programmi di ogni genere. Il tutto a L. 2.900.000 il materiale è in perfette condizioni per informazioni telefonare o scrivere a:
Abburrà Davide - Via S. Martino, 69 - 12032 Barge (CN) - Tel. 0175/926232

Vendo **ZX80** sinclair mai usato completo di alimentatore manuali originali ed in italiano a L. 100.000.
Righi Mauro - Via G. Leopardi, 26 - 22077 Olgiate Comasco - Tel. 031/944239

Ricevuto recentissimamente regalo indesiderato, disposto cedere immediatamente **stampante HP82143A** per HP41C/CV a L. 500.000 (valore commerciale L. 640.000+I.V.A).
Fasano Amedeo - Via Reg. Sagittario - 20090 Segrate (MI) - Tel. 02/2130331

Vendo **Apple II 48K** video 12" 2 drives 8" interf. per IBM compat. (tot. 1 MB in linea) stampante anadex con capacità grafica 140 CPS scheda super "R" term. con programmi bollettazione corrieri - tutto a L. 6.500.000.
Enzo Borghi - Via F.lli Cervi, 5/1 - 42100 Reggio E. - Tel. 0522/74941

Vendo **valigetta** con **Sinclair ZX80** + nuova ROM da 8k + tastiera uguale alo ZX81 + espansione RAM da 16K + alimentatore - completo di tutti i cavi necessari ai vari collegamenti e di manuali + 2 libri per lo ZX80/81 il tutto a L. 300.000.
Battini Alessandro - Via Dante Alighieri, 9 - 24040 Castel Rozzone (BG) - Tel. 0363/83293

Vendo per realizzo **ZX81** + espansione 16K + stampante sinclair tutto nuovissimo garanzie non spedite omaggio cassetta con 40 programmi il tutto a sole L. 450.000.
Lepore Luigi - Via Mameli, 22 - 03039 Sora (FR) - Tel. 832310

Vendo su cassetta i seguenti **programmi** in offerta Alien-b4, Match, Astro invaders, Astro Per, Bets, Simon, Serpentone, il costo della cassetta è di L. 35.000 più spese postal. Spedizione in contrassegno.
Se con l'ordine inviate una vostra cassetta il prezzo sarà di L. 33.000 più spese postali.
Di Gilio Antonio - Via Monte Cervino, 1 - 30030 Favaro Veneto (VE) - Tel. 041/611259

Vendo **programmi BASIC** su Superbrain legge 373 completa, dimensionamento impianti idrici foganti riscaldamento, condizionamento ambienti, calcolo trasmittanze strutture edilizie. Programmi su Floppy dysk disposto corso addestramento.
Mascello Vincenzo - Via C. Rosalba, 46/F Scala L- 70125 Bari - Tel. 518175

Vendo **software CP/M.** Compilatori: BASIC 80, C BASIC, CB80, Cobol, Fortran, Pascal, Macro 80; utility: Word Star, Word Master, CalcStar, Z51D, TExwRiter, ecc. ecc. Su dischi 8 pollici singola densità o 5 pollici per Apple.
Bossi Maurizio - Via Illirico, 11 - 20133 Milano - Tel. (02) 723662

Vendo **programmi di topografia** per Apple eccezionale programma di celerimensura con collegamento automatico di più stazioni e di disegno mappa a curve di livello su C739.
Costa Filippo - Via Peculio, 7 - 94014 Nicosia (Enna) - Tel. 0935/47701

Vendo **software ZX81**: disassembler, snake, poker, wordy, spazio 1999 e tanti altri. Inoltre vendo ZX81 16 KRAM con garanzia, manuale, cavi, alimentatore, libro "Guida al Sinclair", e tanti programmi in omaggio.
Prezzo L. 320.000. Telefonare tra le 21.00 e le 22.00.
Palmese Alessandro - Tel. 06/9180936

Software

Cambio **programmi** VIC 20 impossibile descriverli, inutile catalogo. A chi manda cassetta con programmi mando cassetta di programmi, alcuni opera personale inedita. Statisticamente qualcuno dovrebbe interessare.
Ferrero Federico - Via Virgilio, 4 - 43100 Parma - Tel. 494458

Vendo programmi **gestionali per CBM** serie 4000 e 8000 cambio dispongo di numerosi programmi gestionali e di utilità rispondo a tutti.
Nuvoli Giovanni - Via Ulumos, 3 - 07018 Pozzomaggiore - Tel. 079/801276

Software: consulenza part-time Assembler microprocessori. Telefonare o scrivere.
Invernizzi Andrea - Via Roma, 60 - 20089 Caggiano (MI) - Tel. 9085096

Vendo **software CAD/CAM** americano per Apple 2 DOS 33 con Visicalc. (17 programmi per creare interattivamente disegni salvarli su disco richiamarli aggiornarli e presentarli prospetticamente a video) telefonare di sabato.
Porro Giampiero - Via Zezio, 69 - 22100 Como - Tel. 031/273068

Vendo **gioco ROM Alien** per il computer VIC 20, cambio programmi, esperienze.
Muratori Attilio - Via Roma - 47100 Forlì - Tel. 64221

Cambio **programmi** per Apple II.
Citarella Claudio - Via Parroco Federico, 41 - 80045 Pompei (NA) - Tel. 081/8632946

Programma strutturato per analisi di edifici in cemento armato (telai e/o mensole) sollecitati da carichi permanenti e/o distorsioni termiche e forze sismiche determinate con analisi sismica statica o dinamica, Apple II.
Viola Giovanni - Via Pietragrossa, 1 - 66100 Chieti - Tel. 0871/67477

Vendo e *cambio* per Atari 400-800 molti **programmi gioco** originali americani su cassetta o disco.
Guidotti Marcello - Via Cutilia, 27 - 00183 Roma - 06/778896

Vendo **programmi e schemi** elettrici per ZX80, su richiesta viene inviato il listino prezzi su richiesta si sviluppano programmi per ZX80.
Marziali Marco - Via Calzaivoli, 4 - 50122 Firenze - Tel. 055/284835

Cerco **possessori DAI** (possibilmente in Savona o Genova) per scambio diretto di esperienze - sono disposto a partecipare a un gruppo di acquisto di software originale belga di alto livello di cui dispongo già documentazione: giochi e utilities scrivere, oppure telefonare sab.-dom.
Paolo Siccardo - Via Brignoni, 5/15 - 17100 Savona - Tel. (019) 24526

Cambio - *vendo* **programmi per Apple II**: giochi e utilità.
Benoni Andrea - V.lo Cieco Agnello, 3 - 37100 Verona - Tel. 045/595280

Da spedire a: SINCLUB Sperimentare - Via Dei lavoratori, 124 - 20092 Cinisello B.

PER SAPERNE DI PIU'

Nome e Cognome ______________________ Tel. ____________

Via ______________________ Città ____________ CAP ____________

Professione ______________________

Sei proprietario di un computer Sinclair? NO ☐ SI ☐ ZX80 ☐ ZX81 ☐ SPECTRUM ☐

Sei già socio di un Sinclair Club? SI ☐ NO ☐ (Se si comunicaci la denominazione e l'indirizzo)

Denominazione ______________________

Indirizzo ______________________

Responsabile Club ______________________

Bit LIBRI — Gruppo Editoriale Jackson

Vuoi ordinare dei libri? Spedisci questo tagliando a: Gruppo Editoriale Jackson Via Rosellini, 12 - 20124 Milano.

Nome Cognome

Indirizzo

Cap. Città Provincia

Partita I.V.A. (indispensabile per le aziende)

Si richiede l'emissione della fattura □

Inviatemi i seguenti libri:

Codice Libro	Quantità	Codice Libro	Quantità	Codice Libro	Quantità	Codice Libro	Quantità	Codice Libro	Quantità

□ Pagherò al postino il prezzo indicato + L. 2.000 per contributo fisso spese di spedizione.

□ Allego assegno n° .... di L. ....

□ Non abbonato □ Abbonato sconto 10% □ l'Elettronica □ Elettronica Oggi □ Automazione Oggi □ Elektor
□ Informatica Oggi □ Computerworld □ Bit □ Personal Software □ Strumenti Musicali □ Videogiochi

Data .... Firma ....

Bit LIBRI — Gruppo Editoriale Jackson

Vuoi ordinare dei libri? Spedisci questo tagliando a: Gruppo Editoriale Jackson Via Rosellini, 12 - 20124 Milano.

Nome Cognome

Indirizzo

Cap. Città Provincia

Partita I.V.A. (indispensabile per le aziende)

Si richiede l'emissione della fattura □

Inviatemi i seguenti libri:

Codice Libro	Quantità	Codice Libro	Quantità	Codice Libro	Quantità	Codice Libro	Quantità	Codice Libro	Quantità

□ Pagherò al postino il prezzo indicato + L. 2.000 per contributo fisso spese di spedizione.

□ Allego assegno n° .... di L. ....

□ Non abbonato □ Abbonato sconto 10% □ l'Elettronica □ Elettronica Oggi □ Automazione Oggi □ Elektor
□ Informatica Oggi □ Computerworld □ Bit □ Personal Software □ Strumenti Musicali □ Videogiochi

Data .... Firma ....

Bit BORSA

Sei un lettore di BIT e vuoi entrare in contatto con tutti gli altri lettori per comprare, cambiare o vendere il tuo sistema o qualche particolare accessorio? Spedisci questo tagliando a: Gruppo Editoriale Jackson, Via Rosellini, 12 - 20124 Milano.

□ COMPRO □ CAMBIO □ VENDO

Nome .... Cognome ....

Via .... C.A.P. ....

Città .... Tel. ....

Bit LO SPAZIO DEL RIVENDITORE

Sei un rivenditore di personal computer e vuoi entrare in contatto più diretto con tutti i lettori di BIT? Spedisci questo tagliando a: Gruppo Editoriale Jackson, Via Rosellini, 12 - 20124 Milano

DITTA O NEGOZIO

VIA

CITTA' CAP.

TEL.

MARCHE RAPPRESENTATE

SPECIALIZZATO IN ...

ALTRE NOTIZIE

FIRMA